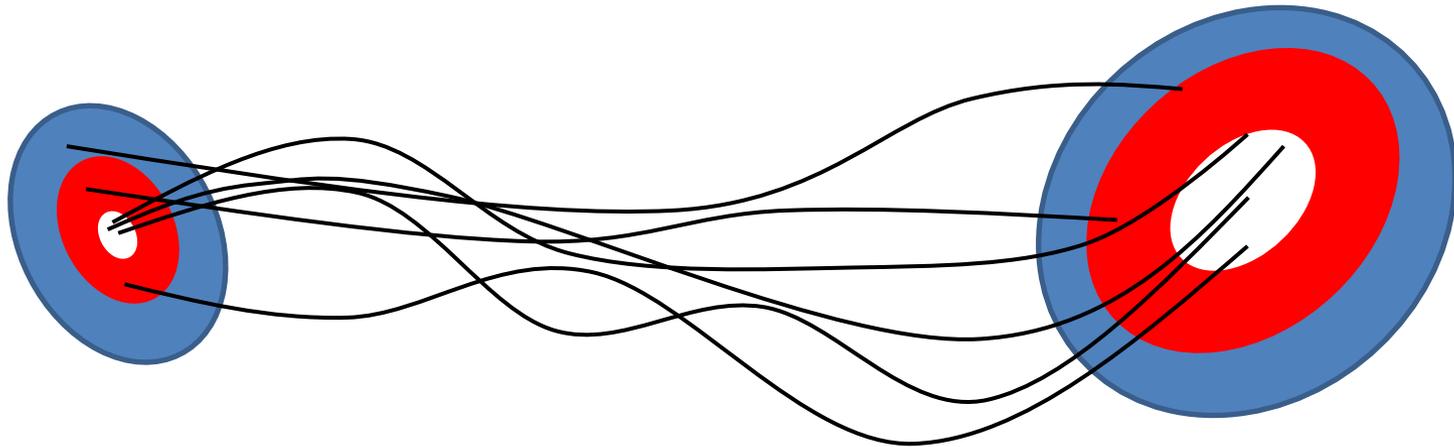


# The Ensemble Kalman filter



## Part I: Theory

Dr Sanita Vetra-Carvalho

(Based on notes by Dr Alison Flower and Dr Ross Bannister)

# Recap of problem we wish to solve

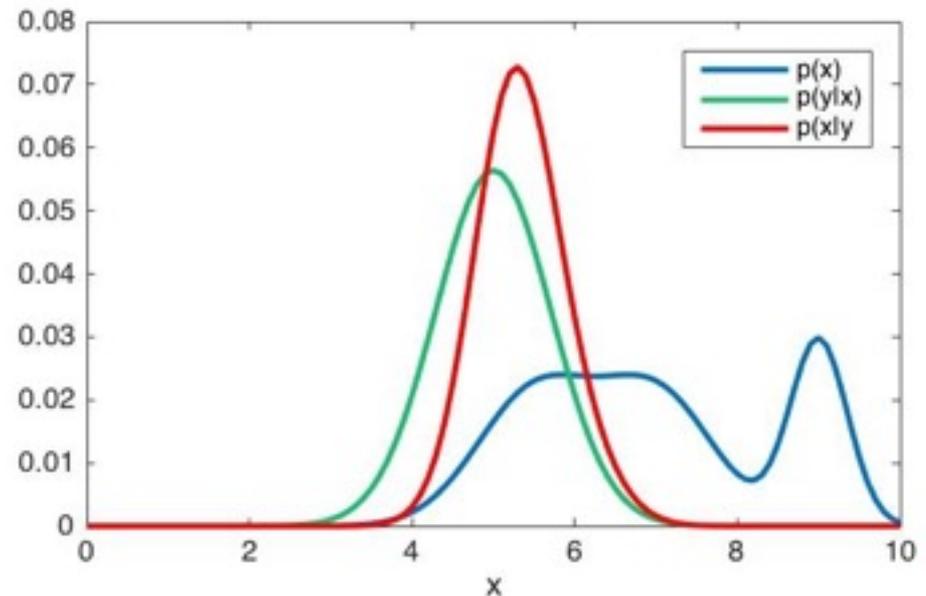
- Given **prior knowledge** of the state of a system and a set of **observations**, we wish to estimate the state of the system at a given time, called **posterior** or **analysis**.
- Bayes' theorem states

$$p(x|y) \propto p(x)p(y|x)$$

**An example: rainfall amount in a given grid box.**

*A-priori* we are unsure if there will be moderate or heavy rainfall. *The observation* only gives probability to the rainfall being moderate.

Applying Bayes' theorem we can now be certain that the rainfall was moderate and the uncertainty is reduced compared to both the observations and our a-priori estimate.



No rain      Moderate rain      Heavy rain

Figure: 1D example of Bayes' theorem.

# Reality bites

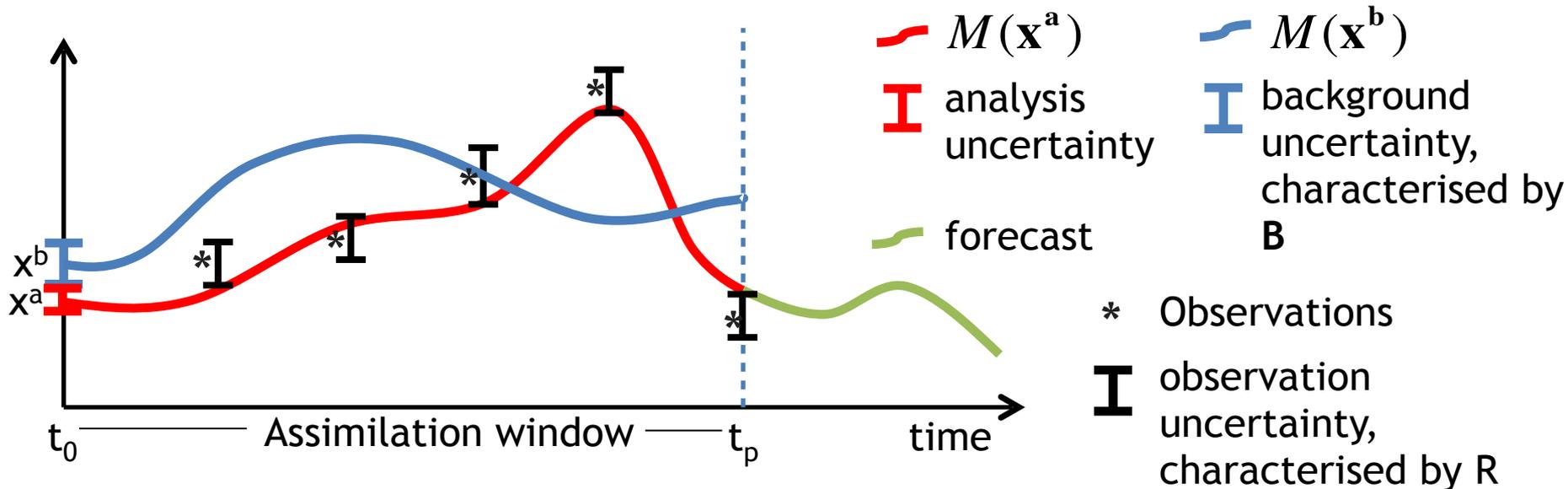
$$p(\mathbf{x}|\mathbf{y}) = \frac{p(\mathbf{x})p(\mathbf{y}|\mathbf{x})}{p(\mathbf{y})}$$

Estimating these pdf's in large dimensional systems is virtually impossible. **Approximate solutions** lead to DA methods:

- **Variational methods:** solves for the *mode* of the posterior.
- **Kalman-based methods:** solve for the *mean* and *covariance* of the posterior.
- **Particle filters:** find a weak (*sample*) *representation* of the posterior pdf.

# Recap of 4DVar

- 4DVar aims to find the most likely state at time  $t_0$ , given an initial estimate,  $x_b$ , and a window of observations.



- $J$  (the cost function) is derived assuming Gaussian error distributions and a perfect model.

$$J(\mathbf{x}_0) = (\mathbf{x}_0 - \mathbf{x}_0^b)^T \mathbf{B}^{-1} (\mathbf{x}_0 - \mathbf{x}_0^b) + \sum_{i=1}^p (\mathbf{y}_i - H(M(\mathbf{x}_0, t_0, t_i)))^T \mathbf{R}_i^{-1} (\mathbf{y}_i - H(M(\mathbf{x}_0, t_0, t_i)))$$

# Recap of 4DVar: why do any differ?

## Advantages

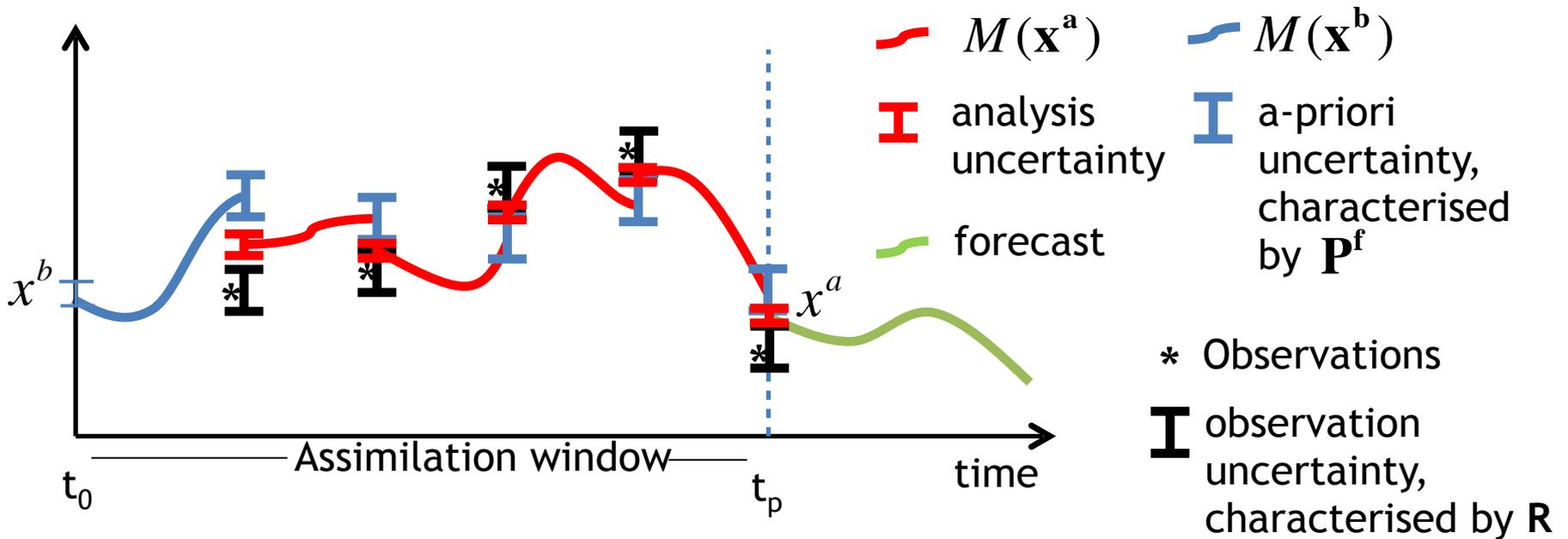
- Gaussian and near-linear assumption makes this an efficient algorithm.
- Minimisation of the cost function is a well posed problem (the B-matrix is designed to be full rank).
- Analysis is consistent with the model (balanced).
- Lots of theory and techniques to modify the basic algorithm to make it a pragmatic method for various applications, e.g. incremental 4DVar, preconditioning, control variable transforms, weak constraint 4DVar...
- Met Office and ECMWF both use methods based on 4DVar for their atmospheric assimilation.

## Disadvantages

- Gaussian assumption is not always valid.
- Relies on the validity of TL and perfect model assumption. This tends to restrict the length of the assimilation window.
- Development of TL model,  $\mathbf{M}$ , and adjoint,  $\mathbf{M}^T$ , is very time consuming and difficult to update as the non-linear model is developed.
- B-matrix is predominately static.

This motivates a different approach...

# Sequential DA



- Instead of assimilating all observations at one time, assimilate them sequentially in time:

$$\begin{aligned}
 p(\mathbf{x}|\mathbf{y}_1, \dots, \mathbf{y}_p) &\propto p(\mathbf{x}|\mathbf{y}_1, \dots, \mathbf{y}_{p-1})p(\mathbf{y}_p|\mathbf{x}) \\
 &\propto p(\mathbf{x}|\mathbf{y}_1, \dots, \mathbf{y}_{p-2})p(\mathbf{y}_{p-1}|\mathbf{x})p(\mathbf{y}_p|\mathbf{x}) \\
 &\vdots \\
 &\propto p(\mathbf{x})p(\mathbf{y}_1|\mathbf{x})\dots p(\mathbf{y}_p|\mathbf{x})
 \end{aligned}$$

# The Kalman filter

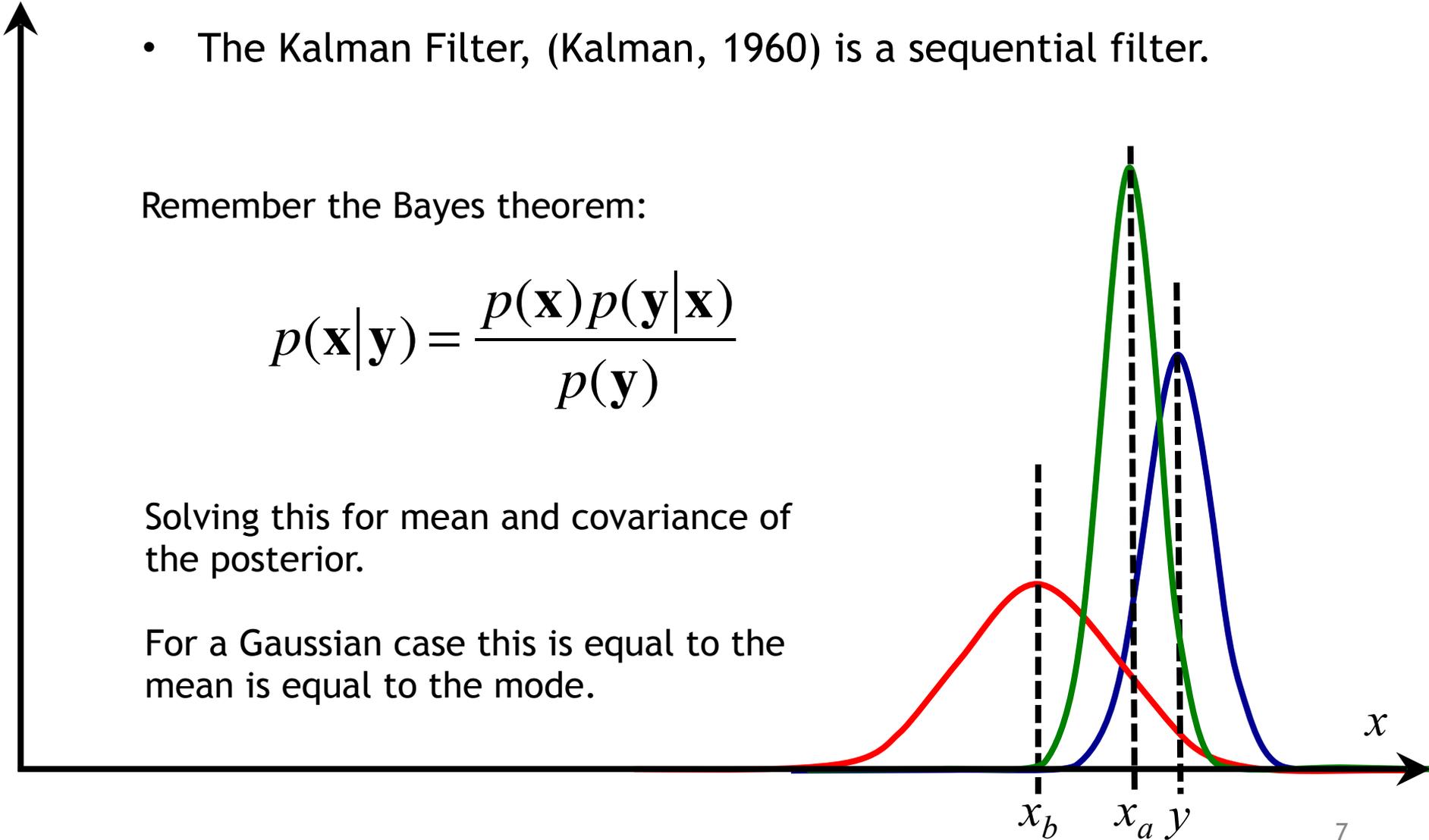
- The Kalman Filter, (Kalman, 1960) is a sequential filter.

Remember the Bayes theorem:

$$p(\mathbf{x}|\mathbf{y}) = \frac{p(\mathbf{x})p(\mathbf{y}|\mathbf{x})}{p(\mathbf{y})}$$

Solving this for mean and covariance of the posterior.

For a Gaussian case this is equal to the mean is equal to the mode.



# The Kalman filter

In Gaussian case the posterior becomes:

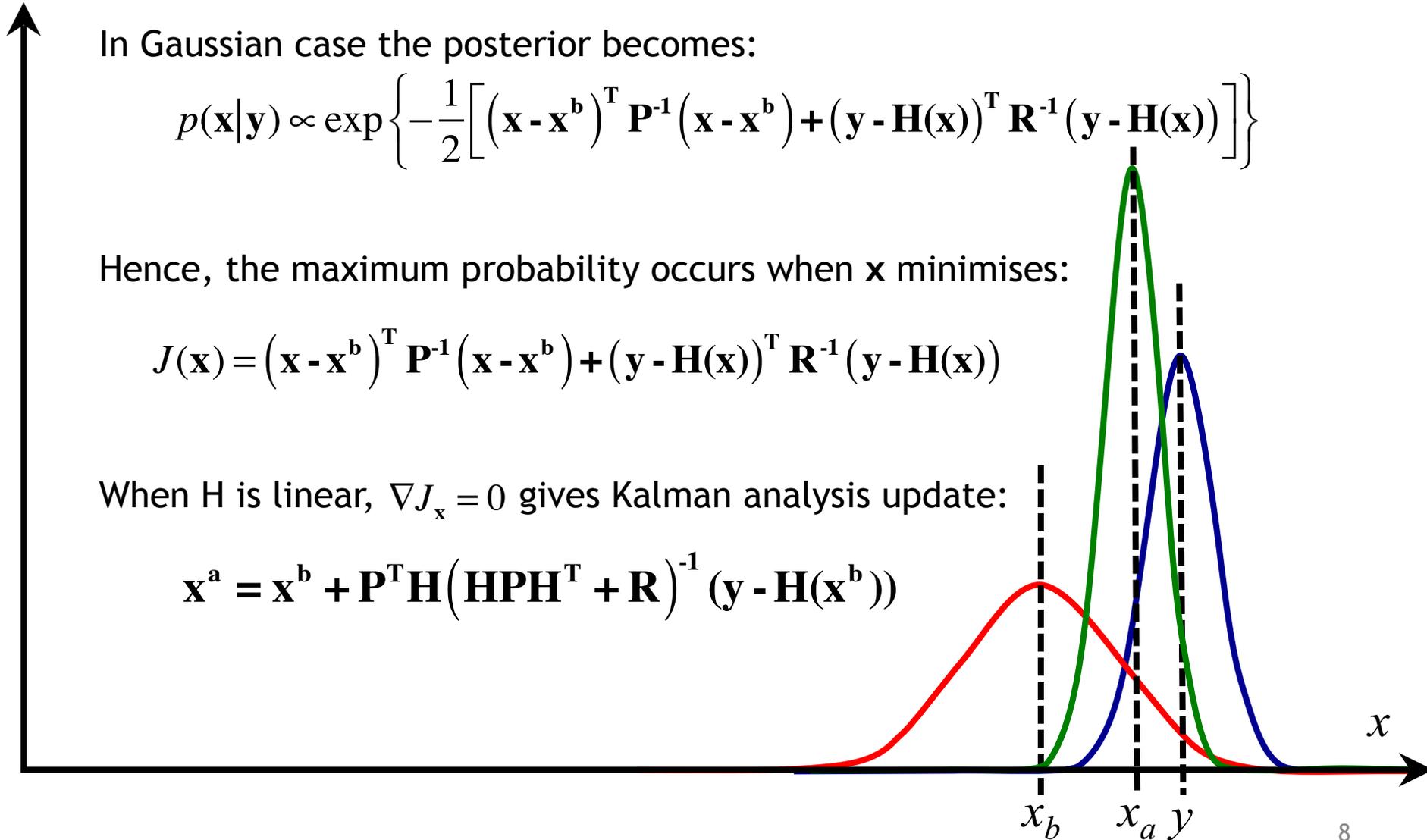
$$p(\mathbf{x}|\mathbf{y}) \propto \exp \left\{ -\frac{1}{2} \left[ (\mathbf{x} - \mathbf{x}^b)^T \mathbf{P}^{-1} (\mathbf{x} - \mathbf{x}^b) + (\mathbf{y} - \mathbf{H}(\mathbf{x}))^T \mathbf{R}^{-1} (\mathbf{y} - \mathbf{H}(\mathbf{x})) \right] \right\}$$

Hence, the maximum probability occurs when  $\mathbf{x}$  minimises:

$$J(\mathbf{x}) = (\mathbf{x} - \mathbf{x}^b)^T \mathbf{P}^{-1} (\mathbf{x} - \mathbf{x}^b) + (\mathbf{y} - \mathbf{H}(\mathbf{x}))^T \mathbf{R}^{-1} (\mathbf{y} - \mathbf{H}(\mathbf{x}))$$

When  $\mathbf{H}$  is linear,  $\nabla J_{\mathbf{x}} = 0$  gives Kalman analysis update:

$$\mathbf{x}^a = \mathbf{x}^b + \mathbf{P}^T \mathbf{H} (\mathbf{H} \mathbf{P} \mathbf{H}^T + \mathbf{R})^{-1} (\mathbf{y} - \mathbf{H}(\mathbf{x}^b))$$



# The Kalman filter algorithm

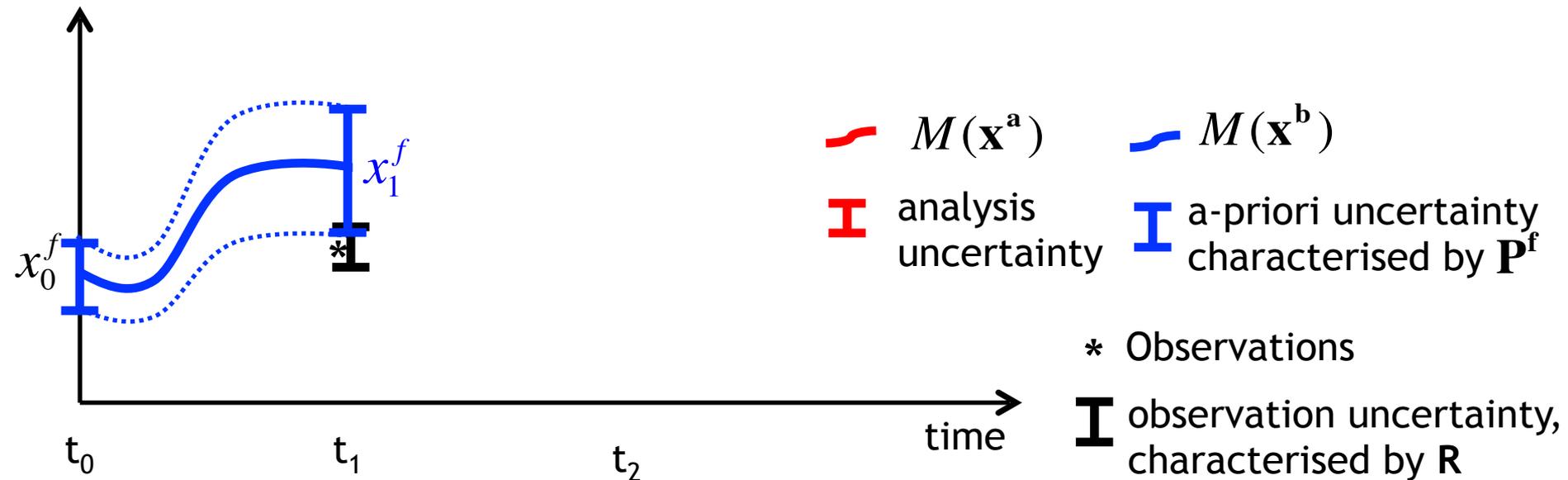
## Prediction or forecast step:

- Evolve the mean state from *to next* observation,

$$\mathbf{x}_n^f = \mathbf{M}\mathbf{x}_{n-1}^a$$

- Evolve the covariance allowing for model error,  $\eta \sim N(0, Q)$

$$\mathbf{P}_n^f = \mathbf{M}\mathbf{P}_{n-1}^a\mathbf{M} + \mathbf{Q}$$



# The Kalman filter algorithm

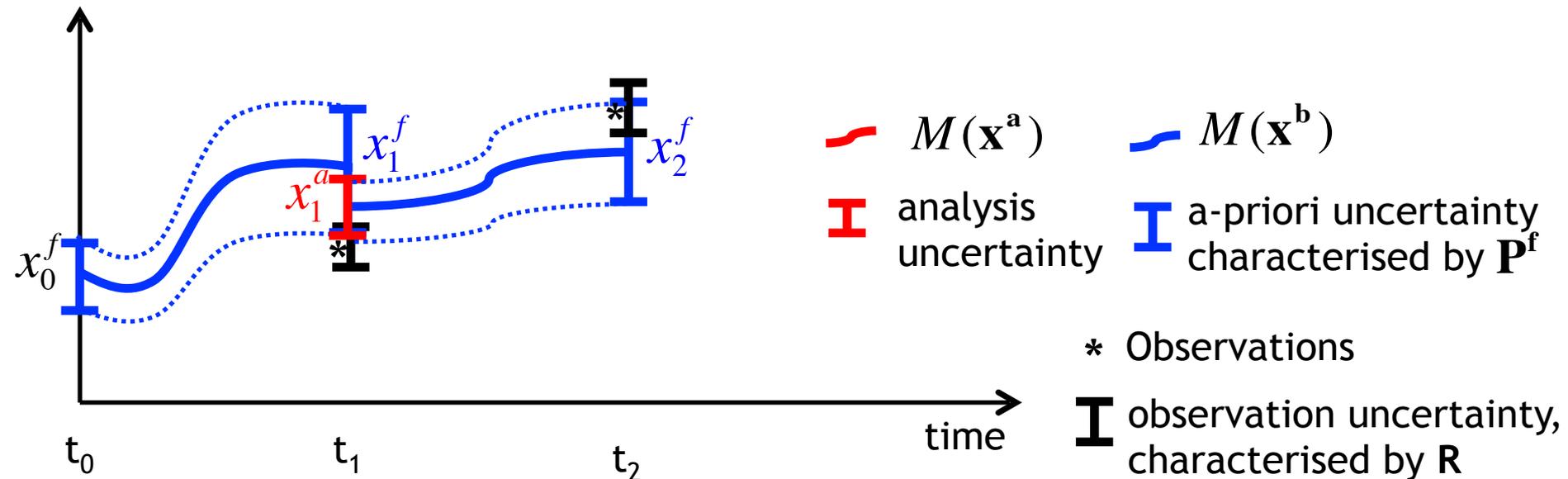
## Analysis or update step:

- Compute Kalman Gain,

$$\mathbf{K}_n = \mathbf{P}_n^f \mathbf{H}^T (\mathbf{H} \mathbf{P}_n^f \mathbf{H}^T + \mathbf{R})^{-1}$$

- Update, mean state and associated covariances,

$$\mathbf{x}_n^a = \mathbf{x}_n^f + \mathbf{K}_n (\mathbf{y}_n - \mathbf{H} \mathbf{x}_n^f) \quad \mathbf{P}_n^a = (\mathbf{I} - \mathbf{K}_n \mathbf{H}) \mathbf{P}_n^f$$



# The Kalman filter algorithm

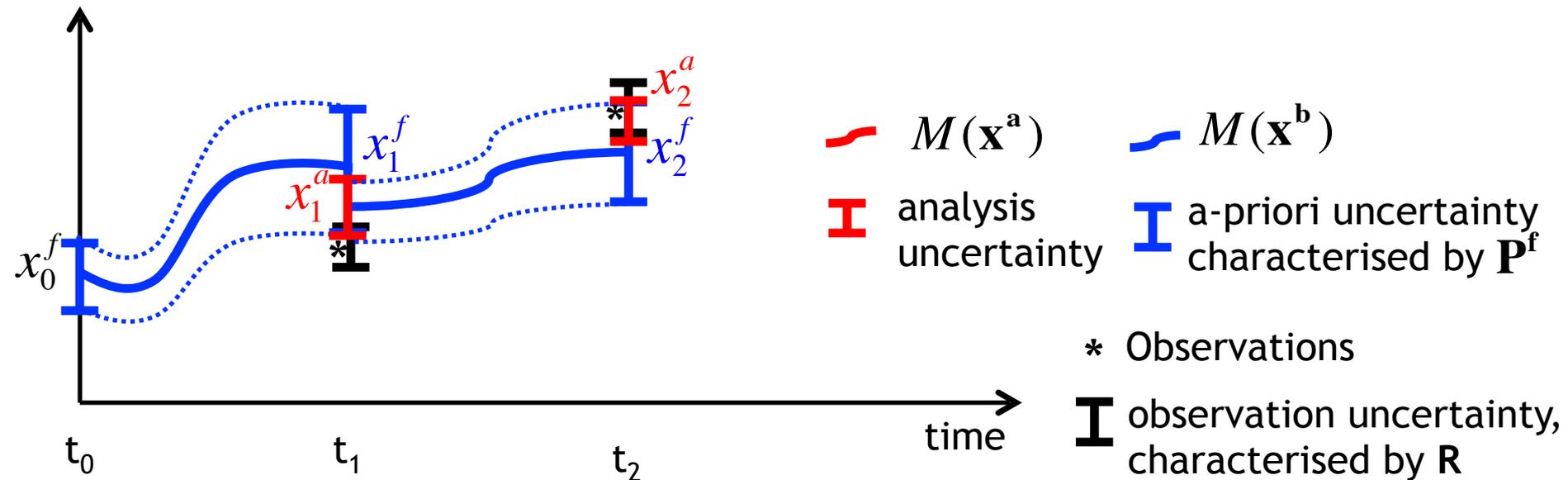
## Analysis or update step:

- Compute Kalman Gain,

$$\mathbf{K}_n = \mathbf{P}_n^f \mathbf{H}^T (\mathbf{H} \mathbf{P}_n^f \mathbf{H}^T + \mathbf{R})^{-1}$$

- Update, mean state and associated covariances,

$$\mathbf{x}_n^a = \mathbf{x}_n^f + \mathbf{K}_n (\mathbf{y}_n - \mathbf{H} \mathbf{x}_n^f) \quad \mathbf{P}_n^a = (\mathbf{I} - \mathbf{K}_n \mathbf{H}) \mathbf{P}_n^f$$



# Motivation for the ensemble Kalman filter (EnKF)

Majority of dynamical models of the environment are not linear. **How can we apply the idea of the KF to non-linear models?**

**Initial solution:** The Extended Kalman Filter (Grewal and Andrews (2008)).

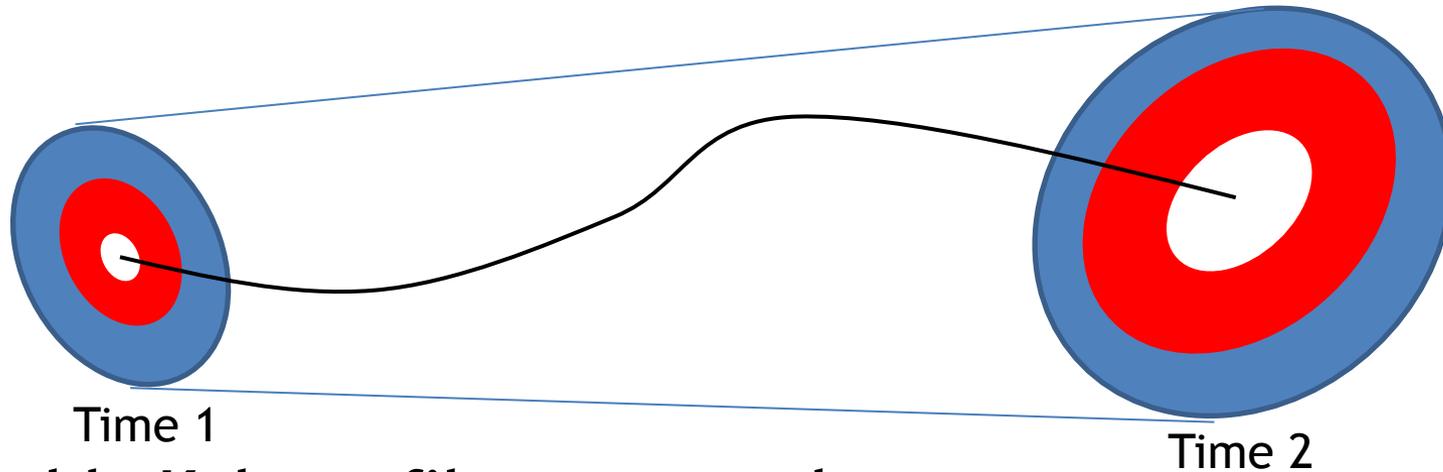
While, the EKF can be applied to non-linear models and/or non-linear observation operators by linearising around the model state, it did not solve the other **major disadvantage of the KF:**

- *the huge and often impossible computational requirements associated with the storage and forward integration of the forecast error covariance matrix,  $\mathbf{P}^f$ .*

Further, the EKF requires the Tangent Linear and adjoint model to propagate the covariance matrix, which as in variational methods are difficult to develop and expensive to maintain.

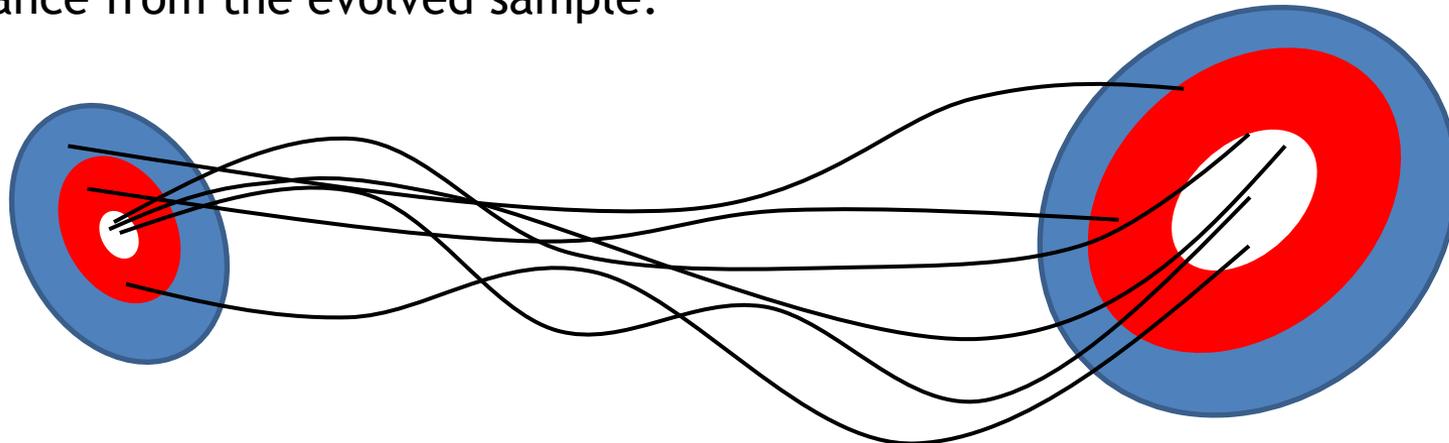
## Extended Kalman filter approach

Explicitly evolve the mean and covariances forward in time using  $M$ ,  $\mathbf{M}$  and  $\mathbf{M}^T$ .



## Ensemble Kalman filter approach

Evolve each state forward in time using  $M$ , then estimate the mean and covariance from the evolved sample.



# Ensemble notation

Given a state vector containing all the state variables,  $\mathbf{X}_i^{\mathbf{a},\mathbf{f}}$ , we define:

- ensemble matrix as,

$$\mathbf{X}^{\mathbf{a},\mathbf{f}} = \left[ \mathbf{X}_1^{\mathbf{a},\mathbf{f}}, \mathbf{X}_2^{\mathbf{a},\mathbf{f}}, \dots, \mathbf{X}_{N_e}^{\mathbf{a},\mathbf{f}} \right]^T \in \mathfrak{R}^{N_x \times N_e}. \quad (1)$$

- ensemble mean as,

$$\bar{\mathbf{X}}^{\mathbf{a},\mathbf{f}} = \frac{1}{N_e} \sum_{i=1}^{N_e} \mathbf{X}_i^{\mathbf{a},\mathbf{f}} \in \mathfrak{R}^{N_x}. \quad (2)$$

- ensemble perturbation matrix as,

$$\mathbf{X}'^{\mathbf{a},\mathbf{f}} = \left[ \mathbf{X}_1^{\mathbf{a},\mathbf{f}} - \bar{\mathbf{X}}^{\mathbf{a},\mathbf{f}}, \mathbf{X}_2^{\mathbf{a},\mathbf{f}} - \bar{\mathbf{X}}^{\mathbf{a},\mathbf{f}}, \dots, \mathbf{X}_{N_e}^{\mathbf{a},\mathbf{f}} - \bar{\mathbf{X}}^{\mathbf{a},\mathbf{f}} \right]^T \in \mathfrak{R}^{N_x \times N_e}. \quad (3)$$

Where, superscripts  $a, f$  stand for analysis and forecast, respectively.

# Ensemble notation

The ensemble estimate of the forecast error covariance matrix is given by,

$$\mathbf{P}^{\mathbf{a},\mathbf{f}} \approx \mathbf{P}_e^{\mathbf{a},\mathbf{f}} = \frac{1}{N_e - 1} \sum_{i=1}^{N_e} (\mathbf{x}_i^{\mathbf{a},\mathbf{f}} - \bar{\mathbf{x}}^{\mathbf{a},\mathbf{f}}) (\mathbf{x}_i^{\mathbf{a},\mathbf{f}} - \bar{\mathbf{x}}^{\mathbf{a},\mathbf{f}})^{\mathbf{T}}$$

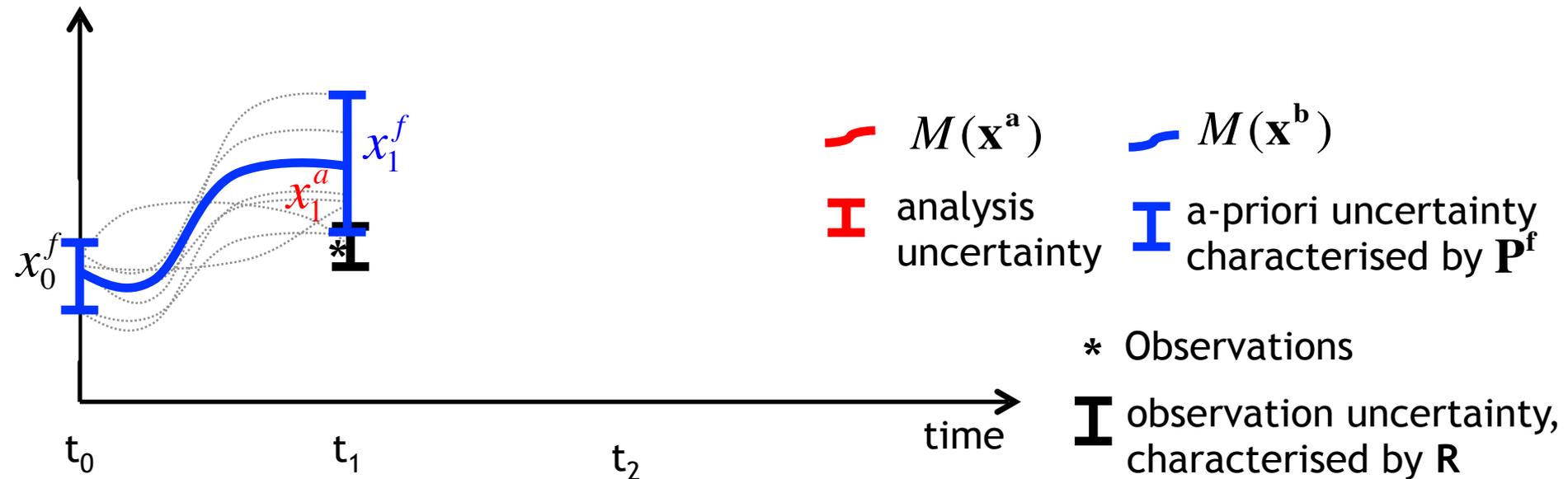
which in matrix form can be written as a matrix outer product

$$\mathbf{P}_e^{\mathbf{a},\mathbf{f}} = \frac{1}{N_e - 1} \mathbf{X}'^{\mathbf{a},\mathbf{f}} (\mathbf{X}'^{\mathbf{a},\mathbf{f}})^{\mathbf{T}}. \quad (5)$$

Time index has been removed for the ease of readability as all the calculations happen at the same time.

# EnKF algorithms

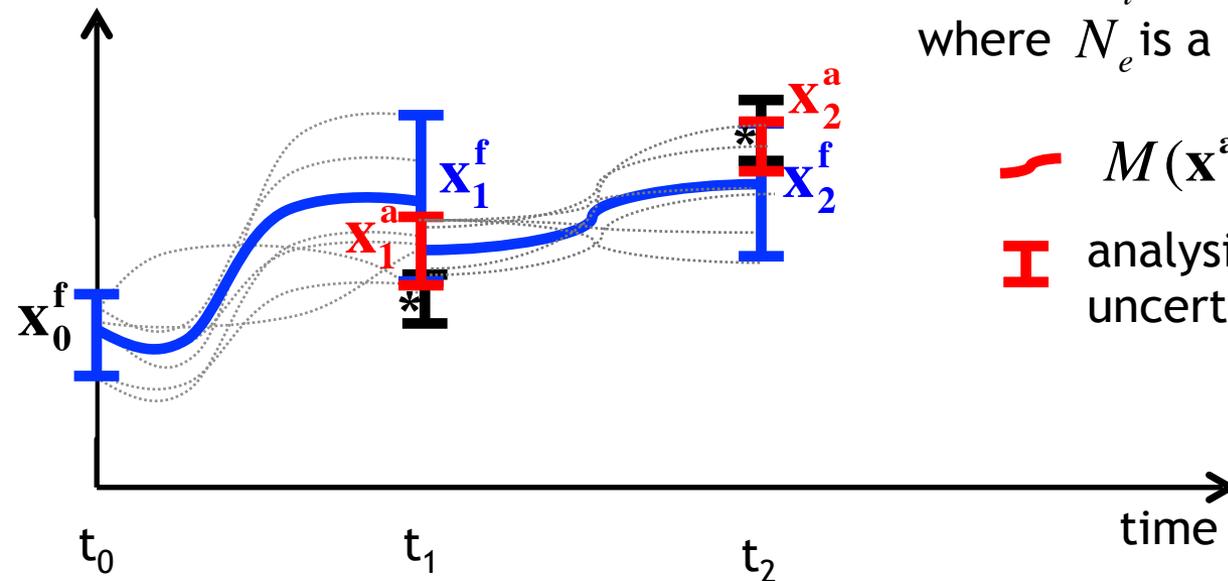
- The EnKF (Envensen 1994) merges the KF theory with the Monte Carlo estimation methods.
- There are many many different flavours of the EnKF, but they all can be represented with the same schematic:



# EnKF algorithms

- The EnKF (Envensen 1994) merges the KF theory with the Monte Carlo estimation methods.
- There are many many different flavours of the EnKF, but they all can be represented with the same schematic:

**Note:**  $\mathbf{x}_i^{\mathbf{a},\mathbf{f}} = [x_1^{\mathbf{a},\mathbf{f}}, x_2^{\mathbf{a},\mathbf{f}}, \dots, x_{N_e}^{\mathbf{a},\mathbf{f}}]$   
 where  $N_e$  is a number of ensemble members



- |   |                              |   |   |
|---|------------------------------|---|---|
|   | $M(\mathbf{x}^{\mathbf{a}})$ |    | $M(\mathbf{x}^{\mathbf{b}})$                                    |
|  | analysis uncertainty         |   | a-priori uncertainty characterised by $\mathbf{P}^{\mathbf{f}}$ |
|   |                              | *   | Observations  |
|   |                              |  | observation uncertainty, characterised by $\mathbf{R}$          |

# EnKF algorithm

All of the EnKF variants have these two common steps:

- *Ensemble forecast* (same for all variants).
- *Ensemble update or analysis* (different for each variant).

Ensemble forecast step is the same for all ensemble Kalman filter methods:

- Forward each ensemble member in time from the previous analysis time to the next observation time using the dynamical model:

$$\mathbf{x}_i^f = M \left( \mathbf{x}_i^a \right) + \eta_i, \quad (4)$$

where

$$i = 1, \dots, N_e$$

and

$$\eta_i \sim N(0, \mathbf{Q}).$$

# EnKF algorithms

The EnKF algorithms can be generalised into two main categories:

- **Stochastic ensemble Kalman filters**, where the use of perturbed observations was introduced simultaneously by Burgers et al. (1998) and Houtekamer and Mitchell (1998) to correct the previously too low spread of the analysis ensemble.
- **Deterministic ensemble Kalman filters**, were designed to avoid the use of perturbed observations, which were found to introduce an additional sampling error into the filter solution. Includes filters:
  - Serial EnSRF (Whitaker and Hamill, 2002)
  - Ensemble Adjustment Kalman Filter (Anderson, 2001)
  - Ensemble Transform Kalman Filter (Bishop et al, 2001), LETKF (Hunt et al, 2007)

# The Stochastic Ensemble Kalman Filter

The original version introduced by Burgers et al. (1998) and Hautekamer and Mitchell (1998) added the perturbation noise onto actual observations. However,

*a much more consistent way is to perturb model predicted observations with the measurement noise, since actual observations already contain errors.*

## Analysis step

- Create perturbed model observations for each ensemble member,  $i$ ,

$$\mathbf{y}_{m,i} = H(\mathbf{x}_i^f) + \boldsymbol{\varepsilon}_{m,i}, \quad (6)$$

where  $\boldsymbol{\varepsilon}_{m,i} \sim N(0, \mathbf{R})$ .

- Compute innovations for each ensemble member,

$$\mathbf{d}_i = \mathbf{y} - \mathbf{y}_{m,i}. \quad (7)$$

# The Stochastic Ensemble Kalman Filter

- Compute Kalman gain,

$$\mathbf{K}_e = \mathbf{P}_e^f \mathbf{H}^T (\mathbf{H} \mathbf{P}_e^f \mathbf{H}^T + \mathbf{R})^{-1}. \quad (8)$$

- Finally, update each ensemble member

$$\mathbf{x}_i^a = \mathbf{x}_i^f + \mathbf{K}_e \mathbf{d}_{m,i}. \quad (9)$$

Note,

- the ensemble forecast error covariances are updated implicitly through the analysis ensemble.
- the forecast error covariance matrix is never explicitly computed, instead we work with ensemble perturbation matrix, remembering that

$$\mathbf{P}_e^f = \frac{1}{N_e - 1} \mathbf{X}'^f (\mathbf{X}'^f)^T.$$

# Summary of the Stochastic EnKF

## Advantages:

- very simple to implement and understand.
- for large ensemble sizes, stochastic filters can handle nonlinearity better than the deterministic filters due to the additional Gaussian observation spread normalising the ensemble update which tends to erase the higher non-Gaussian moments the non-linear error growth has generated (see Lawson and Hansen (2004)).

## Disadvantages:

- due to computational cost we can only afford small ensemble sizes thus offering minimal if any benefit for strongly non-linear problems.
- introduces additional sampling noise.

*This motivates the development of square-root or deterministic forms of the EnKF that do not need to perturb the observations.*

# Deterministic Ensemble Kalman Filters

There is a growing number of deterministic EnKFs all varying in how the *transform matrix*,  $\mathbf{T}$ , is calculated in the analysis step,

$$\mathbf{X}^a = \mathbf{X}^f \mathbf{T}. \quad (10)$$

The idea of the deterministic EnKFs is to create an updated ensemble with covariance consistent with the analysis covariance from KF,

$$\mathbf{P}_e^a = (\mathbf{I} - \mathbf{K}_e \mathbf{H}) \mathbf{P}_e^f. \quad (11)$$

Thus, using eqn. (5) we can write (11) in the square root form,

$$\mathbf{X}'^a (\mathbf{X}'^a)^T = \left( \mathbf{I} - \mathbf{X}'^f (\mathbf{H} \mathbf{X}'^f)^T \left( \mathbf{H} \mathbf{X}'^f (\mathbf{H} \mathbf{X}'^f)^T + (N_e - 1) \mathbf{R} \right)^{-1} \mathbf{H} \right) \mathbf{X}'^f (\mathbf{X}'^f)^T,$$

or defining model observation perturbations as

$$\mathbf{S} = \mathbf{H} \mathbf{X}'^f \quad (12)$$

in a simpler form

$$\mathbf{X}'^a (\mathbf{X}'^a)^T = \left( \mathbf{I} - \mathbf{X}'^f \mathbf{S}^T (\mathbf{S} \mathbf{S}^T + (N_e - 1) \mathbf{R})^{-1} \mathbf{H} \right) \mathbf{X}'^f (\mathbf{X}'^f)^T.$$

# Deterministic Ensemble Kalman Filters

Hence,

$$\mathbf{X}'^a (\mathbf{X}'^a)^T = \mathbf{X}'^f \left( \mathbf{I} - \mathbf{S}^T (\mathbf{S}\mathbf{S}^T + (N_e - 1)\mathbf{R})^{-1} \mathbf{S} \right) (\mathbf{X}'^f)^T.$$

If we define innovation matrix as

$$\mathbf{F} = \mathbf{S}\mathbf{S}^T + (N_e - 1)\mathbf{R}, \quad (13)$$

then we see that we need to find the square root of

$$\mathbf{T}\mathbf{T}^T = \mathbf{I} - \mathbf{S}^T \mathbf{F}^{-1} \mathbf{S} \quad (14)$$

to find analysis ensemble perturbation matrix.

# Ensemble Weight Matrix

- The matrix  $\mathbf{T}$  is chosen such that
$$\begin{aligned}\mathbf{P}_e^a &= \mathbf{X}'^a (\mathbf{X}'^a)^T \\ &= \mathbf{X}'^f \mathbf{T} (\mathbf{X}'^f \mathbf{T})^T \\ &\approx (\mathbf{I} - \mathbf{K}\mathbf{H}) \mathbf{P}^f\end{aligned}$$
- This does not uniquely define  $\mathbf{T}$  which is why there are so many different variants of the square root filters, e.g. the Ensemble Adjustment Kalman Filter (Anderson (2001), and the Ensemble Transform Kalman Filter Bishop et al. (2001)).
- There are a number of good reviews and comparisons on the growing number of square root filters, e.g. Tippett et al (2003), Nerger (2005, 2012), Vetra-Carvalho et al. (2018).

# Deterministic Ensemble Kalman Filters

In general, we can write the analysis updates for deterministic filters as follows:

- Ensemble analysis mean

$$\bar{\mathbf{x}}^a = \bar{\mathbf{x}}^f + \mathbf{X}'^f \bar{\mathbf{w}}, \quad (16)$$

- Ensemble analysis perturbations

$$\mathbf{X}'^a = \mathbf{X}'^f \mathbf{W}', \quad (17)$$

where  $\bar{\mathbf{w}}$  is a weight vector and  $\mathbf{W}'$  is a weight matrix containing the transform matrix.

The weight matrices are filter specific. For example, next we will summarise a very popular ensemble Kalman filter called Ensemble Transform Kalman Filter (ETKF).

# The Ensemble Transform Kalman Filter

Here we use Morrison-Woodbury identity to rewrite

$$\mathbf{T}\mathbf{T}^T = \left( \mathbf{I} + \frac{1}{N_e - 1} \mathbf{S}^T \mathbf{R}^{-1} \mathbf{S} \right)^{-1}$$

Then  $\mathbf{T}$  is computed from the eigenvalue decomposition of

$$\left( \mathbf{T}\mathbf{T}^T \right)^{-1} = \mathbf{U}\mathbf{\Sigma}\mathbf{U}^T.$$

Hence,

$$\bar{\mathbf{w}} = \frac{1}{\sqrt{N_e - 1}} \mathbf{U}\mathbf{\Sigma}^{-1}\mathbf{U}^T \left( \mathbf{X}'^f \right)^T \mathbf{H}^T \mathbf{R}^{-1} \mathbf{d},$$

$$\mathbf{W}' = \mathbf{U}\mathbf{\Sigma}^{-1/2}\mathbf{U}^T.$$

# The Ensemble Transform Kalman Filter

ETKF was first introduced by Bishop et al. (2001) and later revised by Wang et al. (2004).

The revision by Wang et al. highlighted that any  $T$  which satisfies the estimate of the analysis error covariance does not necessarily lead to an unbiased analysis ensemble. Taking a simple single-sided square root could lead to implementations with a biased transformation, such that the transformation by  $W'$  would not preserve the ensemble mean.

However, using the symmetric square-root approach this bias is avoided. For more detail on what conditions  $T$  must satisfy for the analysis ensemble to be centered on the mean see Livings et al. (2008).

# Model error

- The ensemble Kalman filter allows for an imperfect model by adding noise at each time step of the model evolution.

$$\mathbf{x}_i(n) = M(\mathbf{x}_i(n-1)) + \eta_i(n), \quad \eta_i(n) \sim N(0, \mathbf{Q})$$

- The matrix  $\mathbf{Q}$  is not explicitly needed in the algorithm, only the effect of the model error in the evolution of the state.
- There have been many different strategies to including model error in the ensemble, based on where you think the source of the error lies. A few examples are
  - **Multiphysics**- different physical models are used in each ensemble member
  - **Stochastic kinetic energy backscatter**- replaces upscale kinetic energy loss due to unresolved processes and numerical integration.
  - **Stochastically perturbed physical tendencies**
  - **Perturbed parameters**
  - Or combinations of the above

# Extending the state vector

The state vector can be extended with the parameters of the model, called *state augmentation*. Through this we can use the ensemble covariance to update values of poorly known parameters.

$$\tilde{\mathbf{x}}_i^{\mathbf{f}} = \left[ x_{1,i}^{\mathbf{f}}, x_{2,i}^{\mathbf{f}}, \dots, x_{N_x,i}^{\mathbf{f}}, \theta_{1,i}^{\mathbf{f}}, \dots, \theta_{N_p,i}^{\mathbf{f}} \right]^{\mathbf{T}} = \left[ \mathbf{x}_i^{\mathbf{f}}, \boldsymbol{\theta}_i^{\mathbf{f}} \right]^{\mathbf{T}}$$

$$\tilde{\mathbf{P}}_e^{\mathbf{f}} = \begin{pmatrix} \mathbf{P}_e^{\mathbf{f}} & \text{cov}(\mathbf{x}^{\mathbf{f}}, \boldsymbol{\theta}^{\mathbf{f}}) \\ \text{cov}(\boldsymbol{\theta}^{\mathbf{f}}, \mathbf{x}^{\mathbf{f}}) & \text{cov}(\boldsymbol{\theta}^{\mathbf{f}}, \boldsymbol{\theta}^{\mathbf{f}}) \end{pmatrix}$$

The ‘cross-covariance’ carries information from state variables to parameters.

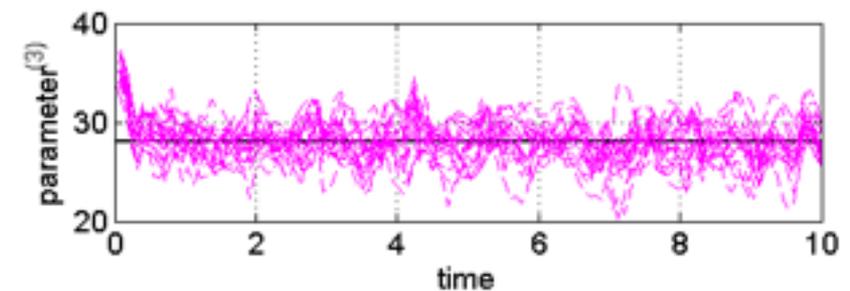
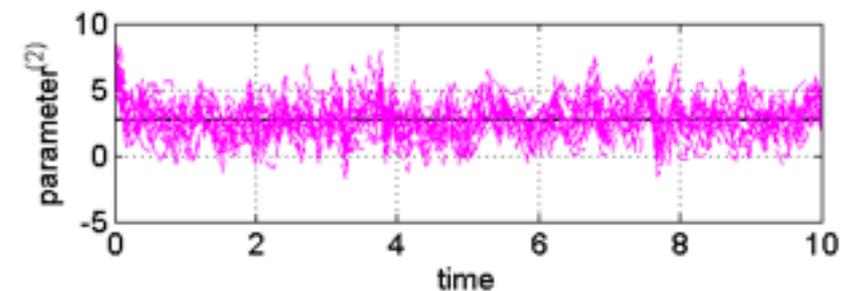
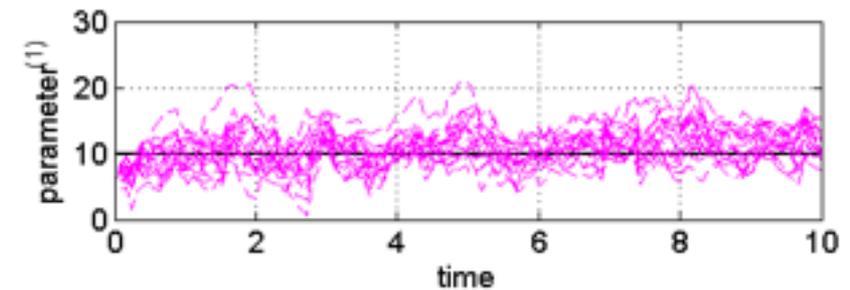
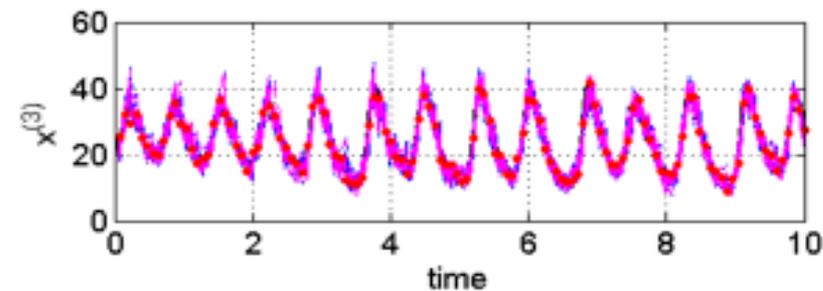
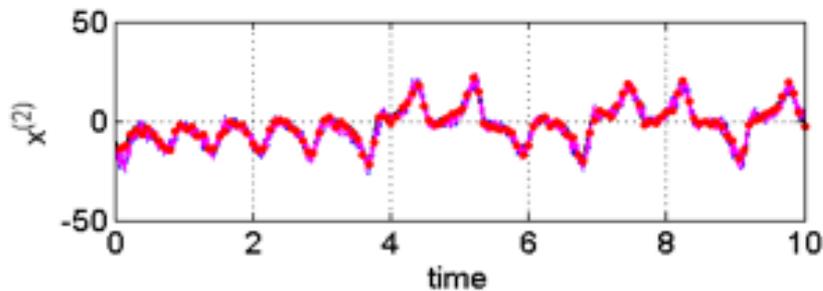
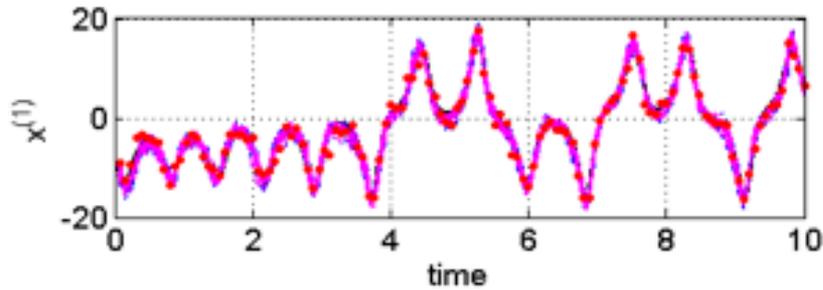
# Extending the state vector

$$\tilde{\mathbf{P}}_e^{\mathbf{f}} = \begin{pmatrix} \mathbf{P}_e^{\mathbf{f}} & \text{cov}(\mathbf{x}^{\mathbf{f}}, \boldsymbol{\theta}^{\mathbf{f}}) \\ \text{cov}(\boldsymbol{\theta}^{\mathbf{f}}, \mathbf{x}^{\mathbf{f}}) & \text{cov}(\boldsymbol{\theta}^{\mathbf{f}}, \boldsymbol{\theta}^{\mathbf{f}}) \end{pmatrix}$$

- During each data-assimilation step the uncertainty in the parameters will decrease.
- Unlike for the model variables, this variance will never increase if we keep the parameters the same between assimilation cycles.
- Often this variance decreases to very small values before the parameters have reached their correct values.
- But very small variance means very small updates, so parameters do not converge to true values.
- Solution is to either perturb parameters with additive noise each model time step, or to include parameters in the inflation.

# Example using Lorenz 1963

Using Lorenz 1963, estimate the values of the state variables and the parameters.



# Summary of the Ensemble Kalman Filter

## Advantages

- The a-priori uncertainty is flow-dependent.
- The code can be developed separately from the dynamical model e.g. EMPIRE and PDAF systems which allow for any model to assimilate observations using ensemble techniques (see <http://www.met.reading.ac.uk/~darc/empire>).
- No need to linearise the model, only linear assumption is that statistics remain close to Gaussian.
- Easy to account for model error.
- Easy to parallelise.

## Disadvantages

- Sensitive to ensemble size. Undersampling can lead to filter divergence. Ideas to mitigate this include localisation and inflation (see next EnKF lecture).
- Assumes Gaussian statistics, for highly non-linear models this may not be a valid assumption (see lecture on particle filters).

# Summary of the Ensemble Kalman Filter

## The different EnKF algorithms

- Many different algorithms exist.
- **Stochastic methods** update each ensemble member separately and then estimate the first two sample moments to give the ensemble mean and covariance.
- **Deterministic methods** update the ensemble simultaneously based on linear/Gaussian theory. May allow for a smaller ensemble than the stochastic methods as avoids some of the sampling error.

## EnKF vs 4DVar

- Each method has its own advantages and disadvantages- there is no clear winner.
- Hybrid methods aim to combine the best bits of both.

# Further reading

- Anderson (2001) An ensemble adjustment filter for data assimilation. *Mon. Weather Rev.*, **129**, 2884-2903.
- Berner et al. (2011) Model uncertainty in a mesoscale ensemble prediction system: Stochastic versus multiphysics representations, *Mon. Weather Rev.*, **139**, 1972-1995.
- Bishop et al. (2001) Adaptive sampling with the ensemble transform Kalman filter. *Mon. Wea. Rev.*, **126**, 1719-1724.
- Evensen (1994) Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *J. Geophys. Res.*, **99(C5)**, 10143-10162.
- Grewal and Andrews (2008) *Kalman Filtering: Theory and Practice using MATLAB*. Wiley, New Jersey.
- Houtekamer and Zhang (2016) Review of the Ensemble Kalman Filter for Atmospheric Data Assimilation. *Mon. Weather Rev.*, **144**, 4489-4532
- Kalman (1960) A new approach to linear filtering and prediction problems. *J. Basic Engineering*, **82**, 32-45.
- Livings et al. (2008) Unbiased ensemble square root filters. *Physica D*. **237**, 1021-1028.
- Tippett et al. (2003) Ensemble Square Root Filters. *Mon. Wea. Rev.*, **131**, 1485-1490.
- Wang et al. (2004) Which Is Better, an Ensemble of Positive-Negative Pairs or a Centered Spherical Simplex Ensemble? *Mon. Wea. Rev.*, **132**, 1590-1605.
- Bannister (2017) A review of operational methods of variational and ensemble-variational data assimilation. *Q. J. R. Meteorol. Soc.*, **143**: 607 - 633.
- Vetra-Carvalho et al. (2018) State-of-the-art stochastic data assimilation methods for high-dimensional non-Gaussian problems. *Tellus A*, <https://doi.org/10.1080/16000870.2018.1445364>