Variational data assimilation I

Background and methods

Ross Bannister

9-12 May 2023, Univ. of Reading

The data assimilation problem

- To combine imperfect data from models, from observations distributed in time and space, exploiting any relevant physical constraints, to produce a more accurate and comprehensive picture of the system as it evolves in time.
- Traditionally we are interested in a state of the system.
- This is just a first moment of the posterior PDF.
- "All models are wrong ..." (George Box)
- "All models are wrong and all observations are inaccurate."

Bayes' Theorem

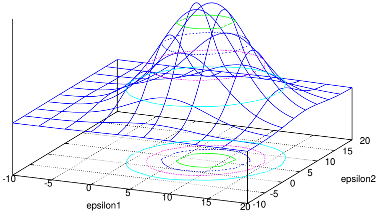
$$p(\mathrm{x}|\mathrm{y}) = \frac{p(\mathrm{x}) \times p(\mathrm{y}|\mathrm{x})}{p(\mathrm{y})}$$

$$\text{posterior distribution} = \frac{\text{prior distribution} \times \text{likelihood}}{\text{normalizing constant}}$$

- Prior distribution: PDF of the state before observations are considered (e.g. PDF of model forecast).
- Likelihood: PDF of observations given that the state is x.
- Posterior: PDF of the state after the obs. have been considered.
- (The "p"s in the above are actually different functions.)

The Gaussian assumption

- PDFs are often described by Gaussians (normal distributions).
- Gaussian PDFs are described by a mean and covariance only.



$$\varepsilon = \mathrm{x} - \langle \mathrm{x} \rangle$$

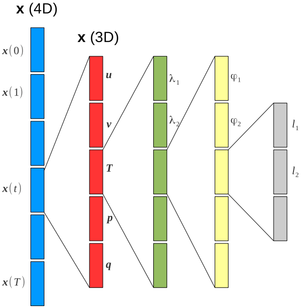
For 1 variable (1D): $x \sim N(\langle x \rangle, \sigma^2)$

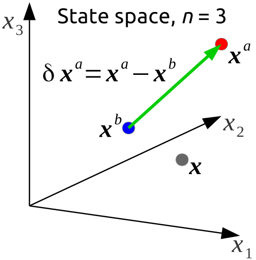
$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp - \frac{(x - \langle x \rangle)^2}{2\sigma^2}$$

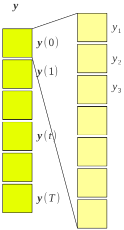
For n variables (nD): $\mathrm{x} \sim N(\langle \mathrm{x} \rangle, \mathrm{C})$

$$p(\mathrm{x}) = \frac{1}{\sqrt{(2\pi)^n \det(\mathrm{C})}} \times \exp - \frac{1}{2} (\mathrm{x} - \langle \mathrm{x} \rangle)^{\mathrm{T}} \mathrm{C}^{-1} (\mathrm{x} - \langle \mathrm{x} \rangle)$$

Meaning of x and y







- Vectors of vectors ...
- x^a analysis; x^b background state; δx increment (perturbation).
- y observations; $y^m = \mathscr{H}(x)$ model observations.
- $\mathscr{H}(x)$ is the observation operator / forward model.
- Sometimes x and y are for only one time (3DVar).
- x-vectors have n elements; y-vectors have p elements.

Back to the Gaussian assumption

Prior: mean x^b, covariance B

$$p(x) = \frac{1}{\sqrt{(2\pi)^n \det(B)}} \exp -\frac{1}{2}\left(x - x^b\right)^T B^{-1}\left(x - x^b\right)$$

Likelihood: mean $\mathscr{H}(x)$, covariance R

$$p(y|x) = \frac{1}{\sqrt{(2\pi)^p \det(R)}} \exp -\frac{1}{2}\left(y - \mathscr{H}(x)\right)^T R^{-1}\left(y - \mathscr{H}(x)\right)$$

Posterior

$$p(x|y) = \frac{p(x) \times p(y|x)}{p(y)} \propto \exp -\frac{1}{2}\Big[\left(x - x^b\right)^T B^{-1}\left(x - x^b\right) + \left(y - \mathscr{H}(x)\right)^T R^{-1}\left(y - \mathscr{H}(x)\right)\Big]$$

Variational DA – the idea

- In Var., we seek a solution that maximizes the posterior probability $p(\mathrm{x}|\mathrm{y})$ (*maximum-a-posteriori*, MAP).
  - This is the most likely state given the observations (and the background), called the analysis, x^{a}.
  - Maximizing $p(\mathrm{x}|\mathrm{y})$ is equivalent to minimizing $-\ln p(\mathrm{x}|\mathrm{y}) \equiv J(\mathrm{x})$ (a least-squares problem).

$$
\begin{aligned}
p(\mathrm{x}|\mathrm{y}) &= C \exp\left\{ -\frac{1}{2} \left[\left(\mathrm{x}-\mathrm{x}^{\mathrm{b}}\right)^{\mathrm{T}} \mathrm{B}^{-1} \left(\mathrm{x}-\mathrm{x}^{\mathrm{b}}\right) \right. \right. \\
&\qquad \left. \left. + \left(\mathrm{y}-\mathscr{H}(\mathrm{x})\right)^{\mathrm{T}} \mathrm{R}^{-1} \left(\mathrm{y}-\mathscr{H}(\mathrm{x})\right) \right] \right\} \\
J(\mathrm{x}) &= -\ln C + \frac{1}{2} \left(\mathrm{x}-\mathrm{x}^{\mathrm{b}}\right)^{\mathrm{T}} \mathrm{B}^{-1} \left(\mathrm{x}-\mathrm{x}^{\mathrm{b}}\right) \\
&\qquad + \frac{1}{2} \left(\mathrm{y}-\mathscr{H}(\mathrm{x})\right)^{\mathrm{T}} \mathrm{R}^{-1} \left(\mathrm{y}-\mathscr{H}(\mathrm{x})\right) \\
&= \text{constant (ignored)} + J_{\mathrm{b}}(\mathrm{x}) + J_{\mathrm{o}}(\mathrm{x})
\end{aligned}
$$

Exercises – practise the 'short hand' algebra

- $\mathrm{u}^{\mathrm{T}}\mathrm{v}$ (product of $1 \times n$ and $n \times 1$ vectors [an inner product], result is 1×1 [a scalar])

$$\begin{pmatrix} u_1 \\ \vdots \\ u_n \end{pmatrix}^{\mathrm{T}} \begin{pmatrix} v_1 \\ \vdots \\ v_n \end{pmatrix} = \begin{pmatrix} u_1 & \cdots & u_n \end{pmatrix} \begin{pmatrix} v_1 \\ \vdots \\ v_n \end{pmatrix} = u_1 v_1 + \cdots + u_n v_n$$

- $\mathrm{u}^{\mathrm{T}}\mathrm{A}\mathrm{v}$ (product of a $1 \times n$, an $n \times n$ matrix, and a $n \times 1$ vector [an inner product in a particular norm], result is 1×1 [a scalar])

$$\begin{pmatrix} u_1 & \cdots & u_n \end{pmatrix} \begin{pmatrix} A_{11} & \cdots & A_{1n} \\ \vdots & \ddots & \vdots \\ A_{n1} & \cdots & A_{nn} \end{pmatrix} \begin{pmatrix} v_1 \\ \vdots \\ v_n \end{pmatrix} = \begin{pmatrix} u_1 & \cdots & u_n \end{pmatrix} \begin{pmatrix} A_{11} v_1 + \cdots + A_{1n} v_n \\ \vdots \\ A_{n1} v_1 + \cdots + A_{nn} v_n \end{pmatrix}$$

$$u_1 \left[A_{11} v_1 + \cdots + A_{1n} v_n \right] + \ldots + u_n \left[A_{n1} v_1 + \cdots + A_{nn} v_n \right]$$

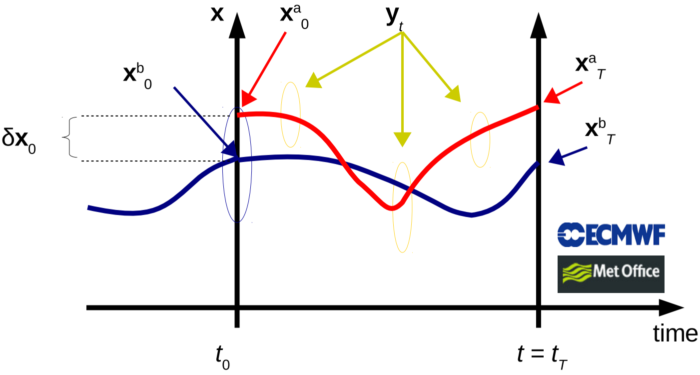
- $\mathrm{u}\mathrm{v}^{\mathrm{T}}$ (product of $n \times 1$ and $1 \times m$ vectors [an outer product], result is $n \times m$ matrix)

$$\begin{pmatrix} u_1 \\ \vdots \\ u_n \end{pmatrix} \begin{pmatrix} v_1 & \cdots & v_m \end{pmatrix} = \begin{pmatrix} u_1 v_1 & \cdots & u_1 v_m \\ \vdots & \ddots & \vdots \\ u_n v_1 & & u_n v_m \end{pmatrix}$$

Four-dimensional Var ("strong constraint" 4DVar)

Aim

To find the 'best' estimate of the true state of the system (analysis), consistent with the observations, the background, and the system dynamics.



Towards a 4DVar cost function

Consider the observation operator in this case:

$$\mathscr{H}(\mathrm{x}) = \mathscr{H}\begin{pmatrix} \mathrm{x}_0 \\ \vdots \\ \mathrm{x}_T \end{pmatrix} = \begin{pmatrix} \mathscr{H}_0(\mathrm{x}_0) \\ \vdots \\ \mathscr{H}_T(\mathrm{x}_T) \end{pmatrix}$$

So the J^{o} is (assume that R is block diagonal):

$$J^{\mathrm{o}} = \frac{1}{2}(\mathrm{y} - \mathscr{H}(\mathrm{x}))^{\mathrm{T}}\mathrm{R}^{-1}(\mathrm{y} - \mathscr{H}(\mathrm{x})) =$$

$$\frac{1}{2}\begin{pmatrix} \mathrm{y}_0 - \mathscr{H}_0(\mathrm{x}_0) \\ \vdots \\ \mathrm{y}_T - \mathscr{H}_T(\mathrm{x}_T) \end{pmatrix}^{\mathrm{T}} \begin{pmatrix} \mathrm{R}_0 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \mathrm{R}_T \end{pmatrix}^{-1} \begin{pmatrix} \mathrm{y}_0 - \mathscr{H}_0(\mathrm{x}_0) \\ \vdots \\ \mathrm{y}_T - \mathscr{H}_T(\mathrm{x}_T) \end{pmatrix}$$

$$= \frac{1}{2}\sum_{i=0}^{T}(\mathrm{y}_i - \mathscr{H}_i(\mathrm{x}_i))^{\mathrm{T}}\mathrm{R}_i^{-1}(\mathrm{y}_i - \mathscr{H}_i(\mathrm{x}_i))$$

subject to the strong constraint $\mathrm{x}_{i+1} = \mathscr{M}_i(\mathrm{x}_i)$

The 4DVar cost function ('full 4DVar')

$$\text{Let } (\mathrm{a})^{\mathrm{T}} \mathrm{A}^{-1} (\mathrm{a}) \equiv (\mathrm{a})^{\mathrm{T}} \mathrm{A}^{-1} (\bullet)$$

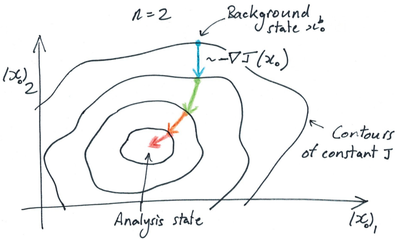
$$\begin{aligned} J(\mathrm{x}) &= \frac{1}{2}\left(\mathrm{x}_0 - \mathrm{x}_0^{\mathrm{b}}\right)^{\mathrm{T}} \mathrm{B}_0^{-1} (\bullet) + \frac{1}{2}(\mathrm{y} - \mathscr{H}(\mathrm{x}))^{\mathrm{T}} \mathrm{R}^{-1} (\bullet) \\ &= \frac{1}{2}\left(\mathrm{x}_0 - \mathrm{x}_0^{\mathrm{b}}\right)^{\mathrm{T}} \mathrm{B}_0^{-1} (\bullet) + \frac{1}{2}\sum_{i=0}^{T} (\mathrm{y}_i - \mathscr{H}_i(\mathrm{x}_i))^{\mathrm{T}} \mathrm{R}_i^{-1} (\bullet) \end{aligned}$$

subject to the strong constraint $\mathrm{x}_{i+1} = \mathscr{M}_i(\mathrm{x}_i)$

- $\mathrm{x}_0^{\mathrm{b}}$ a-priori (background) state at t_0; x_i state at t_i; y_i obs at t_i.
- $\mathscr{H}_i(\mathrm{x}_i)$ observation operator at t_i.
- B_0 background error covariance matrix at t_0.
- R_i observation error covariance matrix at t_i.
- Ultimately J is a fn of x_0 as $\mathrm{x}_i = \mathscr{M}_{i-1}(\mathscr{M}_{i-2}(\cdots \mathscr{M}_0(\mathrm{x}_0)))$.

How to minimize this ('full 4DVar') cost function?

Minimize $J(\mathrm{x}_0)$ iteratively



Use the gradient of J at each iteration:

$$\mathrm{x}_0^{k+1} = \mathrm{x}_0^k + \alpha \nabla J(\mathrm{x}_0^k)$$

The gradient of the cost function

$$\nabla J(\mathrm{x}_0) = \begin{pmatrix} \partial J / \partial [\mathrm{x}_0]_1 \\ \vdots \\ \partial J / \partial [\mathrm{x}_0]_n \end{pmatrix}$$

$-\nabla J$ points in the direction of steepest descent.

Methods: steepest descent (inefficient), conjugate gradient, quasi-Newton (more efficient), ...

The gradient of the cost function (wrt $\mathrm{x}(t_0)$)

Either:

1. Minimise $J(\mathrm{x}_0)$ w.r.t. x_0 with $\mathrm{x}_i = \mathscr{M}_{i-1}(\mathscr{M}_{i-2}(\cdots\mathscr{M}_0(\mathrm{x}_0)))$.
2. Minimise $J(\mathrm{x}) = J(\mathrm{x}_0, \mathrm{x}_1, \ldots, \mathrm{x}_T)$ w.r.t. $\mathrm{x}_0, \mathrm{x}_1, \ldots, \mathrm{x}_T$ subject to the constraint

$$\mathrm{x}_{i+1} - \mathscr{M}_i(\mathrm{x}_i) = 0$$

$$L(\mathrm{x}, \lambda) = J(\mathrm{x}) + \sum_{i=0}^{T-1} \lambda_{i+1}^{\mathrm{T}} (\mathrm{x}_{i+1} - \mathscr{M}_i(\mathrm{x}_i)).$$

Each approach leads to the **adjoint method**

- An efficient means of computing the gradient.
- Uses the linearised/adjoint of $\mathscr{M}_i$ and $\mathscr{H}_i$: $\mathrm{M}_i^{\mathrm{T}}$ and $\mathrm{H}_i^{\mathrm{T}}$ (see next slides).

The adjoint method

Equivalent gradient formula:

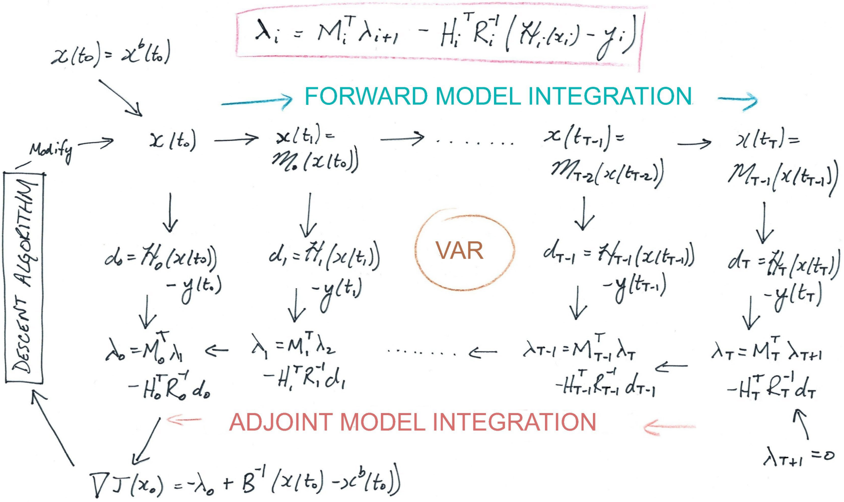
1.

$$
\begin{aligned}
\nabla J \equiv \nabla J(\mathrm{x}_0) &= \nabla J_{\mathrm{b}}(\mathrm{x}_0) + \nabla J_{\mathrm{o}}(\mathrm{x}_0) \\
&= \mathrm{B}_0^{-1}\left(\mathrm{x}_0 - \mathrm{x}_0^{\mathrm{b}}\right) \\
&\quad - \sum_{i=0}^{T} \mathrm{M}_0^{\mathrm{T}} \ldots \mathrm{M}_{i-1}^{\mathrm{T}} \mathrm{H}_i^{\mathrm{T}} \mathrm{R}_i^{-1} \left(\mathrm{y}_i - \mathscr{H}_i(\mathrm{x}_i)\right) \\
\text{where } \mathrm{M}_i &= \partial \mathscr{M}_i(\mathrm{x}_i)/\partial \mathrm{x}_i \text{ and } \mathrm{H}_i = \partial \mathscr{H}_i(\mathrm{x}_i)/\partial \mathrm{x}_i
\end{aligned}
$$

2.

$$
\begin{aligned}
\lambda_{T+1} &= 0 \\
\lambda_i &= \mathrm{H}_i^{\mathrm{T}} \mathrm{R}_i^{-1} \left(\mathrm{y}_i - \mathscr{H}_i(\mathrm{x}_i)\right) + \mathrm{M}_i^{\mathrm{T}} \lambda_{i+1} \\
\lambda_0 &= -\nabla J_{\mathrm{o}} \\
\therefore \nabla J &= \nabla J_{\mathrm{b}} + \nabla J_{\mathrm{o}} \\
&= \mathrm{B}_0^{-1}\left(\mathrm{x}_0 - \mathrm{x}_0^{\mathrm{b}}\right) - \lambda_0
\end{aligned}
$$

The adjoint method



Simplifications and complications

- The full 4DVar method is expensive and difficult to solve.
- Model $\mathscr{M}_i$ is non-linear.
- Observation operators, $\mathscr{H}_i$ can be non-linear.
- Linear $\mathscr{H} \rightarrow$ quadratic cost function – easy(er) to minimize, $J^{\mathrm{o}} \sim \frac{1}{2}(y - ax)^2 / \sigma_{\mathrm{o}}^2$.
- Non-linear $\mathscr{H} \rightarrow$ non-quadratic cost function – hard to minimize, $J^{\mathrm{o}} \sim \frac{1}{2}(y - f(x))^2 / \sigma_{\mathrm{o}}^2$.
- Later will recognise that models are 'wrong'!

Look for simplifications:

Incremental 4DVar (linearised 4DVar)
3D-FGAT
3DVar

Complications:

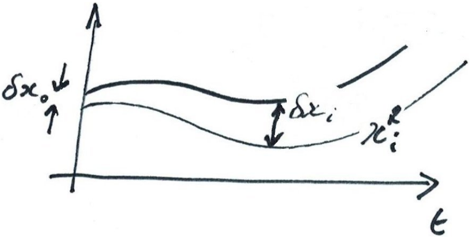
Weak constraint
(imperfect model)

Incremental 4DVar (1)

define reference trajectory: $\mathrm{x}_{i+1}^{\mathrm{R}} = \mathscr{M}_i\left(\mathrm{x}_i^{\mathrm{R}}\right) \qquad \mathrm{y}_i^{\mathrm{mR}} = \mathscr{H}_i\left(\mathrm{x}_i^{\mathrm{R}}\right)$

$$\mathrm{x}_i = \mathrm{x}_i^{\mathrm{R}} + \delta\mathrm{x}_i \qquad \mathrm{x}_0^{\mathrm{b}} = \mathrm{x}_0^{\mathrm{R}} + \delta\mathrm{x}_0^{\mathrm{b}}$$

$$\begin{aligned}
\mathrm{x}_{i+1} &= \mathscr{M}_i(\mathrm{x}_i) = \mathscr{M}_i\left(\mathrm{x}_i^{\mathrm{R}} + \delta\mathrm{x}_i\right) \\
\mathrm{x}_{i+1}^{\mathrm{R}} + \delta\mathrm{x}_{i+1} &\approx \mathscr{M}_i\left(\mathrm{x}_i^{\mathrm{R}}\right) + \mathrm{M}_i\delta\mathrm{x}_i \qquad \delta\mathrm{x}_{i+1} \approx \mathrm{M}_i\delta\mathrm{x}_i \\
\mathrm{y}_i^{\mathrm{m}} &= \mathscr{H}_i(\mathrm{x}_i) = \mathscr{H}_i\left(\mathrm{x}_i^{\mathrm{R}} + \delta\mathrm{x}_i\right) \\
\mathrm{y}_i^{\mathrm{mR}} + \delta\mathrm{y}_i^{\mathrm{m}} &\approx \mathscr{H}_i\left(\mathrm{x}_i^{\mathrm{R}}\right) + \mathrm{H}_i\delta\mathrm{x}_i \qquad \delta\mathrm{y}_i^{\mathrm{m}} \approx \mathrm{H}_i\delta\mathrm{x}_i
\end{aligned}$$



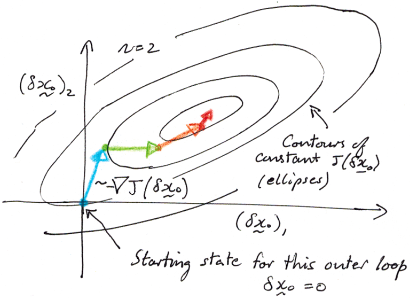
Incremental 4DVar (2)

$$
\begin{aligned}
J(\delta \mathrm{x}_0) &= \frac{1}{2}\left(\delta \mathrm{x}_0 - \delta \mathrm{x}_0^{\mathrm{b}}\right)^{\mathrm{T}} \mathrm{B}_0^{-1}(\bullet) + \\
&\quad \frac{1}{2}\sum_{i=0}^{T}\left(\mathrm{y}_i - \mathscr{H}_i(\mathrm{x}_i^{\mathrm{R}}) - \mathrm{H}_i \delta \mathrm{x}_i\right)^{\mathrm{T}} \mathrm{R}_i^{-1}(\bullet) \\
\delta \mathrm{x}_i &\approx \mathrm{M}_{i-1}\mathrm{M}_{i-2}\ldots\mathrm{M}_0 \delta \mathrm{x}_0
\end{aligned}
$$

- Initially set reference to background, $\mathrm{x}_0^{\mathrm{R}} = \mathrm{x}_0^{\mathrm{b}}$.
- 'Inner loop': iterations to find $\delta \mathrm{x}_0^{\mathrm{a}} = \mathrm{argmin}\, J(\delta \mathrm{x}_0)$ (use adjoint method).
- 'Outer loop': iterate $\mathrm{x}_0^{\mathrm{R}} \to \mathrm{x}_0^{\mathrm{R}} + \delta \mathrm{x}_0^{\mathrm{a}}$
- Inner loop is exactly quadratic (e.g. has a unique minimum).
- Inner loop can be simplified (lower res., simplified physics).

How to minimize this ('incremental 4DVar') cost function?

Minimize $J(\delta x_0)$ iteratively



Use the gradient of J at each iteration:

$$\delta x_0^{k+1} = \delta x_0^k + \alpha \nabla J(\delta x_0^k)$$

The gradient of the cost function

$$\nabla J(\delta x_0) = \begin{pmatrix} \partial J / \partial [\delta x_0]_1 \\ \vdots \\ \partial J / \partial [\delta x_0]_n \end{pmatrix}$$

$-\nabla J$ points in the direction of steepest descent.

Methods: steepest descent (inefficient), conjugate gradient, quasi-Newton (more efficient), . . .

Simplification 1: incremental 3D-FGAT

- Three dimensional variational data assimilation with first guess (i.e. x_i^R) is computed at the appropriate time.
- Simplification is that $M_i \to I$, i.e. $\delta x_i = M_{i-1} \ldots M_0 \delta x_0 \to \delta x_0$:

$$
\begin{aligned}
J^{3DFGAT}(\delta x_0) &= \frac{1}{2}\left(\delta x_0 - \delta x_0^b\right)^T B_0^{-1}(\bullet) + \\
&\quad \frac{1}{2}\sum_{i=-T/2}^{T/2}\left(y_i - \mathscr{H}_i(x_i^R) - H_i \delta x_0\right)^T R_i^{-1}(\bullet).
\end{aligned}
$$

- Note the centring of the assimilation window about t_0 (to reduce the impact of the 3D-FGAT approximation).

Simplification 2: incremental 3DVar

- This has no time dependence within the assimilation window.
- Not used (these days "3DVar" really means 3D-FGAT).

$$
\begin{aligned}
J^{\mathrm{3DVar}}(\delta x_0) &= \frac{1}{2}\left(\delta x_0 - \delta x_0^{\mathrm{b}}\right)^{\mathrm{T}} B_0^{-1} (\bullet) + \\
&\quad \frac{1}{2} \sum_{i=-T/2}^{T/2} \left(y_i - \mathscr{H}_i(x_0^{\mathrm{R}}) - H_i \delta x_0\right)^{\mathrm{T}} R_i^{-1} (\bullet)
\end{aligned}
$$

- But note: 3DVar is not an approx. if all obs. in this cycle are at $t = 0$ (no time index $t = 0$). For $x^{\mathrm{R}} = x^{\mathrm{b}}$:

$$
\begin{aligned}
J^{\mathrm{3DVar}}(\delta x) &= \frac{1}{2}\delta x^{\mathrm{T}} B^{-1} \delta x + \frac{1}{2}\left(y - \mathscr{H}(x^{\mathrm{b}}) - H\delta x\right)^{\mathrm{T}} R^{-1} (\bullet) \\
\text{Setting } \nabla J^{\mathrm{3DVar}} &= B^{-1}\delta x - H^{\mathrm{T}} R^{-1}\left(y - \mathscr{H}(x^{\mathrm{b}}) - H\delta x\right) = 0 \\
\text{Gives } x^{\mathrm{a}} = x^{\mathrm{b}} + \delta x &= x^{\mathrm{b}} + \left(B^{-1} + H^{\mathrm{T}} R^{-1} H\right)^{-1} H^{\mathrm{T}} R^{-1}\left(y - \mathscr{H}(x^{\mathrm{b}})\right) \\
\text{As the Kalman Filter!} &= x^{\mathrm{b}} + BH^{\mathrm{T}}\left(R + HBH^{\mathrm{T}}\right)^{-1}\left(y - \mathscr{H}(x^{\mathrm{b}})\right)
\end{aligned}
$$

Reminder: the Kalman Filter

$$\begin{aligned} \mathrm{x}_t^{\mathrm{a}} &= \mathrm{x}_t^{\mathrm{f}} + \mathrm{K}_t \left(\mathrm{y}_{\mathrm{t}} - \mathscr{H}_t(\mathrm{x}_t^{\mathrm{f}})\right) \\ \mathrm{P}_t^{\mathrm{a}} &= (\mathrm{I} - \mathrm{K}_{\mathrm{t}} \mathrm{H}_t) \mathrm{P}_t^{\mathrm{f}} \\ \mathrm{K}_t &= \mathrm{P}_t^{\mathrm{f}} \mathrm{H}_t^{\mathrm{T}} \left(\mathrm{R}_t + \mathrm{H}_t \mathrm{P}_t^{\mathrm{f}} \mathrm{H}_t^{\mathrm{T}}\right)^{-1} \end{aligned}$$

$\longleftarrow$

$$\begin{aligned} \mathrm{x}_{t+1}^{\mathrm{f}} &= \mathscr{M}_t(\mathrm{x}_t^{\mathrm{a}}) \\ \mathrm{P}_{t+1}^{\mathrm{f}} &= \mathrm{M}_t \mathrm{P}_t^{\mathrm{a}} \mathrm{M}_t^{\mathrm{T}} + \mathrm{Q}_t \end{aligned}$$

$$\left(\mathrm{B}^{-1} + \mathrm{H}^{\mathrm{T}} \mathrm{R}^{-1} \mathrm{H}\right) \mathrm{B} \mathrm{H}^{\mathrm{T}}$$

$$= \mathrm{H}^{\mathrm{T}} \mathrm{R}^{-1} \left(\mathrm{R} + \mathrm{H} \mathrm{B} \mathrm{H}^{\mathrm{T}}\right)$$

(S-M-W formula)

$$\begin{aligned} \mathrm{H}_t &= \left.\frac{\partial\left(\mathscr{H}_t(\mathrm{x})\right)}{\mathrm{x}}\right|_{\mathrm{x}=\mathrm{x}_t^{\mathrm{f}}} \\ \mathrm{M}_t &= \left.\frac{\partial\left(\mathscr{M}_t(\mathrm{x})\right)}{\mathrm{x}}\right|_{\mathrm{x}=\mathrm{x}_t^{\mathrm{a}}} \end{aligned}$$

Properties of 4DVar

- Observations are treated at the correct time.
- Use of dynamics means that more information can be obtained from observations.
- Covariance B_0 is implicitly evolved, $B_i = (M_{i-1} \dots M_0) B_0 (M_{i-1} \dots M_0)^{\mathrm{T}}$.
- In practice development of linear and adjoint models is complex.
  - $\mathscr{M}_i$, $\mathscr{H}_i$, M_i, H_i, M_i^{T}, and H_i^{T} are subroutines, and so 'matrices' are usually not in explicit matrix form.

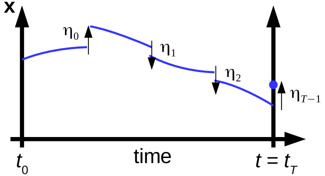
But note

- Standard 4DVar assumes the model is perfect.
- This can lead to sub-optimalities.
- Weak-constraint 4DVar relaxes this assumption.

Weak constraint 4DVar

Modify evolution equation:

$$\begin{aligned} \mathrm{x}_{i+1} &= \mathscr{M}_i(\mathrm{x}_i) + \eta_i \\ \text{where } \eta_i &\sim N(0, \mathrm{Q}_i) \end{aligned}$$



'State formulation' of WC4DVar

$$J^{\mathrm{wc}}_{\mathrm{state}}(\mathrm{x}_0, \ldots, \mathrm{x}_T) = J^{\mathrm{b}} + J^{\mathrm{o}} + \frac{1}{2}\sum_{i=0}^{T-1}(\mathrm{x}_{i+1} - \mathscr{M}_i(\mathrm{x}_i))^{\mathrm{T}}\mathrm{Q}_i^{-1}(\bullet)$$

'Error formulation' of WC4DVar

$$J^{\mathrm{wc}}_{\mathrm{error}}(\mathrm{x}_0, \eta_0 \ldots, \eta_{T-1}) = J^{\mathrm{b}} + J^{\mathrm{o}} + \frac{1}{2}\sum_{i=0}^{T-1}\eta_i^{\mathrm{T}}\mathrm{Q}_i^{-1}\eta_i$$

Implementation of weak constraint 4DVar

- Vector to be determined ('control vector') increases from n in 4DVar to $n + nT$ in WC4DVar.
- The model error covariance matrices, Q_i, need to be estimated. How?
- The 'state' formulation (determine $\mathrm{x}_0, \ldots, \mathrm{x}_T$) and the 'error' formulation (determine $\mathrm{x}_0, \eta_0 \ldots, \eta_{T-1}$) are mathematically equivalent, but can behave differently in practice.
- There is an incremental form of WC4DVar.

Summary of 4DVar

- The variational method forms the basis of many operational weather and ocean forecasting systems, including at ECMWF, the Met Office, Météo-France, etc.
- It allows complicated observation operators to be used (e.g. for assimilation of satellite data).
- It has been very successful.
- Incremental (quasi-linear) versions are usually implemented.
- It requires specification of B_0, the background error cov. matrix, and R_i, the observation error cov. matrix.
- 4DVar requires the development of linear and adjoint models – not a simple task!
- Weak constraint formulations require the additional specification of Q_i.

Selected References

- *Original application of 4DVar*: Talagrand O, Courtier P, Variational assimilation of meteorological observations with the adjoint vorticity equation I: Theory, Q. J. R. Meteorol. Soc. 113, 1311–1328 (1987).
- *Excellent tutorial on Var*: Schlatter TW, Variational assimilation of meteorological observations in the lower atmosphere: A tutorial on how it works, J. Atmos. Sol. Terr. Phys. 62, 1057–1070 (2000).
- *Incremental 4DVar*: Courtier P, Thepaut J-N, Hollingsworth A, A strategy for operational implementation of 4D-Var, using an incremental approach, Q. J. R. Meteorol. Soc. 120, 1367–1387 (1994).
- *High-resolution application of 4DVar*: Park SK, Zupanski D, Four-dimensional variational data assimilation for mesoscale and storm scale applications, Meteorol. Atmos. Phys. 82, 173–208 (2003).
- *Met Office 4DVar*: Rawlins F, Ballard SP, Bovis KJ, Clayton AM, Li D, Inverarity GW, Lorenc AC, Payne TJ, The Met Office global four-dimensional variational data assimilation scheme, Q. J. R. Meteorol. Soc. 133, 347–362 (2007).
- *Weak constraint 4DVar*: Tremolet Y, Model-error estimation in 4D-Var, Q. J. R. Meteorol. Soc. 133, 1267–1280 (2007).
- *Inner and outer loops*: Lawless, Gratton & Nichols, QJRMS, 2005; Gratton, Lawless & Nichols, SIAM J. on Optimization (2007).
- *More detailed survey of variational methods than can be done in this lecture (plus ensemble-variational, hybrid methods)*: Bannister R.N., A review of operational methods of variational and ensemble-variational data assimilation, Q.J.R. Meteor. Soc. 143, 607-633 (2017).