Assimilation Algorithms
Lecture 3: 4D-Var

Sébastien Massart

ECMWF

23 May 2023

A Covid story - part 3

+ 0 h	Vaccine
+ 1 h	36.0°C
+ 2 h	36.0°C
+ 3 h	36.0°C
+ 6 h	36.5°C
+ 7 h	37.0°C
+ 8 h	38.0°C
+ 8 h	Paracetamol
+ 9 h	37.5°C
+ 10 h	37.0°C

Outline

1. Strong Constraint 4D-Var: Derivation
2. Strong Constraint 4D-Var: Calculating the Cost and Gradient
3. The Incremental Method
4. FGAT formulation
5. Weak Constraint 4D-Var
6. Summary

Outline

Strong Constraint 4D-Var

- So far, we have tacitly assumed that the observations, analysis and background are all valid at the same time, so that $\mathcal{H}$ includes spatial, but not temporal, interpolation.

- In 4D-Var, we relax this assumption.

- Let's use $\mathcal{G}$ to denote a generalised observation operator that:
  1. Propagates model fields defined at some time t_0 to the (various) times at which the observations were taken.
  2. Spatially interpolates these propagated fields
  3. Converts model variables to observed quantities

- We will use a numerical forecast model to perform the first step.

- Note that, since models integrate forward in time and we do not have an inverse of the forecast model, the observations must be available for times $t_k \geq t_0$, for $k \in \{1, 2, \cdots, K\}$.

Strong Constraint 4D-Var

- Formally, the 4D-Var cost function is identical to the 3D-Var cost function — we simply replace $\mathcal{H}$ by $\mathcal{G}$:

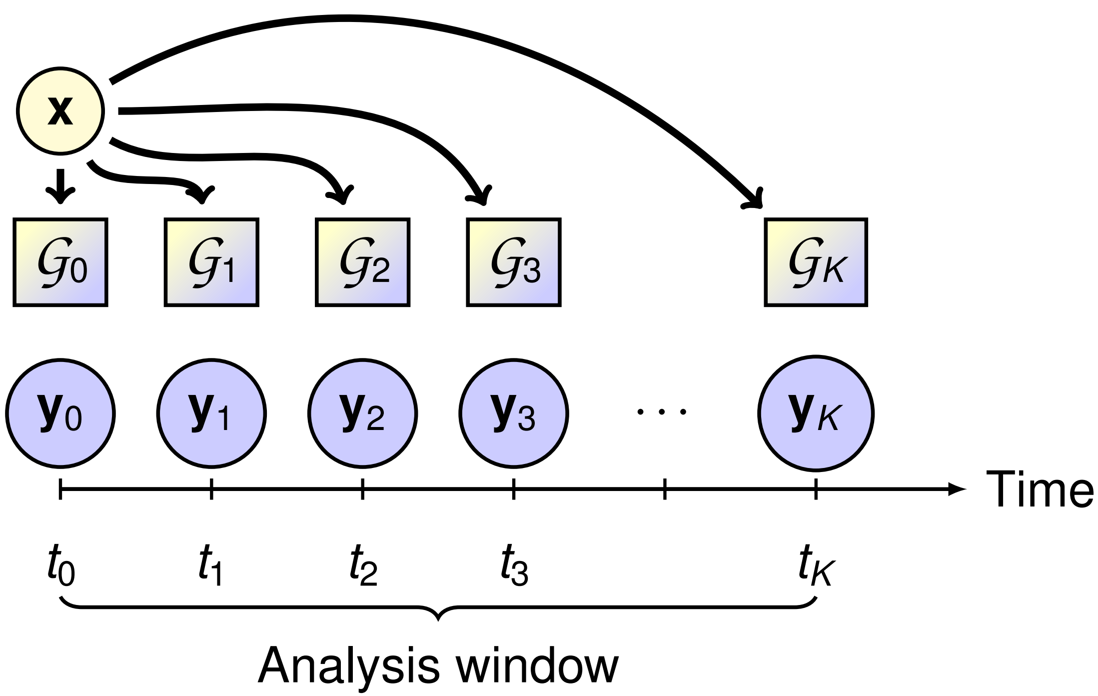
$$J(\mathbf{x}) = \frac{1}{2}(\mathbf{x} - \mathbf{x}_b)^{\mathrm{T}} \mathbf{P}^{b^{-1}} (\mathbf{x} - \mathbf{x}_b) + \frac{1}{2}\left[\mathbf{y} - \mathcal{G}(\mathbf{x})\right]^{\mathrm{T}} \mathbf{R}^{-1} \left[\mathbf{y} - \mathcal{G}(\mathbf{x})\right]$$

- However, it makes sense to group observations into sub-vectors of observations, $\mathbf{y}_k$, that are valid at the same time, t_k.
- It is reasonable (or at least convenient) to assume that observation errors are uncorrelated in time. Then, $\mathbf{R}$ is block diagonal, with blocks $\mathbf{R}_k$ corresponding to the sub-vectors $\mathbf{y}_k$.

$$\mathbf{y} = \begin{pmatrix} \mathbf{y}_0 \\ \vdots \\ \mathbf{y}_k \\ \vdots \\ \mathbf{y}_K \end{pmatrix}, \quad \mathbf{R} = \begin{pmatrix} \mathbf{R}_0 & & & & \\ & \ddots & & \mathbf{0} & \\ & & \mathbf{R}_k & & \\ & \mathbf{0} & & \ddots & \\ & & & & \mathbf{R}_K \end{pmatrix}, \quad \text{and} \quad \mathcal{G}(\mathbf{x}) = \begin{pmatrix} \mathcal{G}_0(\mathbf{x}) \\ \vdots \\ \mathcal{G}_k(\mathbf{x}) \\ \vdots \\ \mathcal{G}_K(\mathbf{x}) \end{pmatrix}$$

Strong Constraint 4D-Var

$$
\begin{aligned}
J(\mathbf{x}) \;=\;& \frac{1}{2}(\mathbf{x}-\mathbf{x}_b)^{\mathrm{T}}\mathbf{P}^{b^{-1}}(\mathbf{x}-\mathbf{x}_b) \\
&+\frac{1}{2}\sum_{k=0}^{K}\left[\mathbf{y}_k-\mathcal{G}_k(\mathbf{x})\right]^{\mathrm{T}}\mathbf{R}_k^{-1}\left[\mathbf{y}_k-\mathcal{G}_k(\mathbf{x})\right]
\end{aligned}
$$

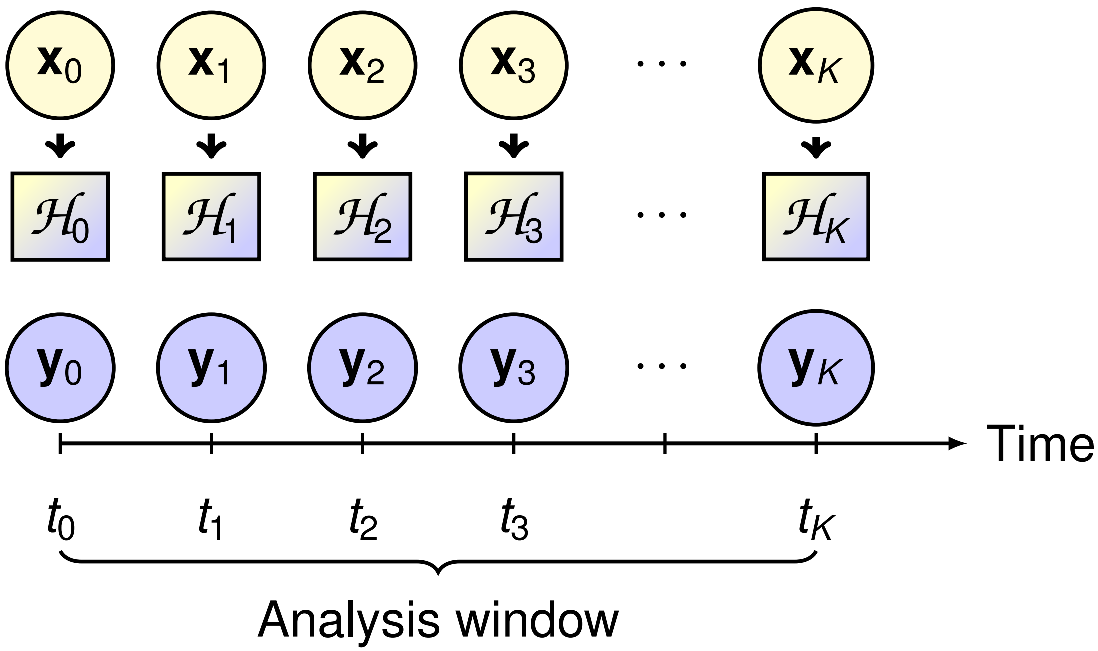


Strong Constraint 4D-Var

- Let us introduce model states $\mathbf{x}_k$, which are defined at times t_k.
- Now, each generalised observation operator can be written as

$$\mathcal{G}_k = \mathcal{H}_k(\mathbf{x}_k)$$

where $\mathcal{H}_k$ represents a spatial interpolation and transformation from model variables to observed variables — i.e. a 3D-Var-style observation operator.



Strong Constraint 4D-Var

- Then, we can write the cost function as:

$$
\begin{aligned}
J(\mathbf{x}_0, \mathbf{x}_1, \cdots, \mathbf{x}_K) = & \frac{1}{2}(\mathbf{x}_0 - \mathbf{x}_b)^{\mathrm{T}} \mathbf{P}^{b^{-1}} (\mathbf{x}_0 - \mathbf{x}_b) \\
& + \frac{1}{2} \sum_{k=0}^{K} \left[\mathbf{y}_k - \mathcal{H}_k(\mathbf{x}_k)\right]^{\mathrm{T}} \mathbf{R}_k^{-1} \left[\mathbf{y}_k - \mathcal{H}_k(\mathbf{x}_k)\right].
\end{aligned}
$$

- We will use a numerical forecast model to propagate model fields from time t_0 to time t_k, for $k \in \{1, 2, \cdots, K\}$:

$$
\begin{aligned}
\mathbf{x}_k &= \mathcal{M}_{t_0 \to t_k}(\mathbf{x}_0) \\
&= \mathcal{M}_{t_{k-1} \to t_k}(\mathbf{x}_{k-1}),
\end{aligned}
$$

  where $\mathcal{M}_{t_0 \to t_k}$ represents an integration of the forecast model from time t_0 to time t_k and $\mathcal{M}_{t_{k-1} \to t_k}$ from time t_{k-1} to time t_k .
- Now, each generalised observation operator can be written as the concatenation of the model $\mathcal{M}_{t_0 \to t_k}$ and the observation operator $\mathcal{H}_k$:

$$
\mathcal{G}_k = \mathcal{H}_k \circ \mathcal{M}_{t_0 \to t_k}.
$$

Strong Constraint 4D-Var

Unconstrained minimisation problem

- We had initially an unconstrained minimisation problem:

$$\mathbf{x}_a = \arg\min_{\mathbf{x}} \left(J(\mathbf{x}) \right)$$

where possibly $\mathbf{x} = (\mathbf{x}_0, \mathbf{x}_1, \cdots \mathbf{x}_K)$

Strong constraint 4D-Var

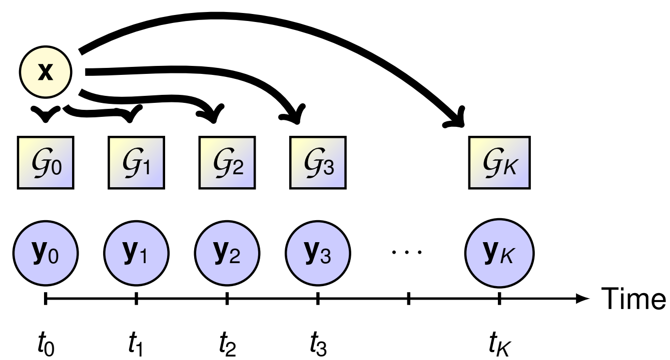
- By introducing the forecast model $\mathcal{M}_{t_0 \to t_k}$ from time t_0 to time t_k that links $\mathbf{x}_k$ to $\mathbf{x}_0$, we have converted the minimisation problem into a problem with strong constraints:

$$\begin{aligned} \mathbf{x}_a &= \arg\min_{\mathbf{x}_0} \left(J(\mathbf{x}_0, \mathbf{x}_1, \cdots \mathbf{x}_K) \right) \\ \text{where} \quad \mathbf{x}_k &= \mathcal{M}_{t_0 \to t_k}(\mathbf{x}_0) \quad \text{for } k = 1, 2, \cdots, K \end{aligned}$$

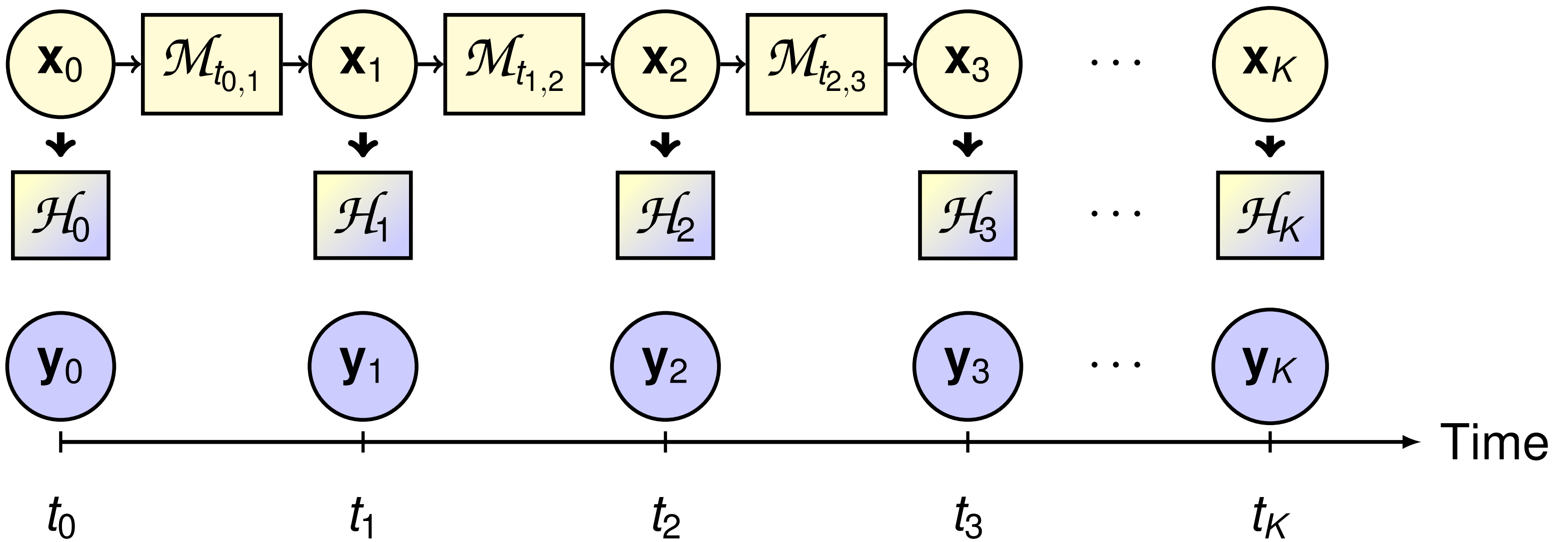
- For this reason, this form of 4D-Var is called strong constraint 4D-Var.

Strong Constraint 4D-Var

Initial formulation



Strong Constraint formulation



Strong Constraint 4D-Var

- When we derived the 3D-Var cost function, we assumed that the observation operator was perfect: $\mathbf{y}_t = \mathcal{H}(\mathbf{x}_t)$.
- In deriving strong constraint 4D-Var, we have not removed this assumption.
- The generalised observation operators, $\mathcal{G}_k$, are assumed to be perfect.
- In particular, since $\mathcal{G}_k = \mathcal{H}_k \circ \mathcal{M}_{t_0 \to t_k}$, this implies that the model is perfect:

$$\mathbf{x}_{t,k} = \mathcal{M}_{t_{k-1} \to t_k}\left(\mathbf{x}_{t,k-1}\right).$$

- This is called the perfect model assumption.

Strong Constraint 4D-Var

$$J(\mathbf{x}_0, \mathbf{x}_1, \cdots \mathbf{x}_k) = \frac{1}{2}(\mathbf{x}_0 - \mathbf{x}_b)^{\mathrm{T}} \mathbf{P}^{b^{-1}} (\mathbf{x}_0 - \mathbf{x}_b) + \frac{1}{2}\sum_{k=0}^{K} \left[\mathbf{y}_k - \mathcal{H}_k(\mathbf{x}_k)\right]^{\mathrm{T}} \mathbf{R}_k^{-1} \left[\mathbf{y}_k - \mathcal{H}_k(\mathbf{x}_k)\right]$$

- ✗ When written in this form, it is clear that 4D-Var determines the analysis state at every gridpoint *and at every time within the analysis window*.
- ✗ I.e., 4D-Var determines a four-dimensional analysis of the available asynoptic data.
- ✗ As a consequence of the perfect model assumption, the analysis corresponds to a trajectory (i.e. an integration) of the forecast model.

Outline

Strong Constraint 4D-Var

- ✗ In general, unconstrained minimisation problems are easier to solve than constrained problems.
- ✗ To minimise the cost function, we write it as a function of $\mathbf{x}_0$:

$$
\begin{aligned}
J(\mathbf{x}_0) \;=\; & \frac{1}{2}(\mathbf{x}_0 - \mathbf{x}_b)^{\mathrm{T}} \mathbf{P}^{b^{-1}} (\mathbf{x}_0 - \mathbf{x}_b) \\
& + \frac{1}{2}\sum_{k=0}^{K} \left[\mathbf{y}_k - \mathcal{G}_k(\mathbf{x}_0)\right]^{\mathrm{T}} \mathbf{R}_k^{-1} \left[\mathbf{y}_k - \mathcal{G}_k(\mathbf{x}_o)\right]
\end{aligned}
$$

- ✗ We avoid repeating integrations of the model with the following algorithm:

$J := \frac{1}{2}(\mathbf{x}_0 - \mathbf{x}_b)^{\mathrm{T}} \mathbf{P}^{b^{-1}} (\mathbf{x}_0 - \mathbf{x}_b)$;

for $k = 0, 1, \cdots, K$ **do**

  /* Add the contribution of time t_k */

  $J := J + \frac{1}{2}\left[\mathbf{y}_k - \mathcal{H}_k(\mathbf{x}_k)\right]^{\mathrm{T}} \mathbf{R}_k^{-1} \left[\mathbf{y}_k - \mathcal{H}_k(\mathbf{x}_k)\right]$;

  /* Integrate the model from time t_k to time t_{k+1} */

  $\mathbf{x}_{k+1} := \mathcal{M}_{t_k \rightarrow t_{k+1}}(\mathbf{x}_k)$;

end

Strong Constraint 4D-Var

- As in 3D-Var, efficient minimisation of the cost function requires us to calculate its gradient.
- Differentiating the unconstrained version of the cost function with respect to $\mathbf{x}_0$ gives:

$$\nabla J(\mathbf{x}_0) = \mathbf{P}^{b^{-1}}(\mathbf{x}_0 - \mathbf{x}_b) - \sum_{k=0}^{K} \mathbf{G}_k^{\mathrm{T}} \mathbf{R}_k^{-1} \left[\mathbf{y}_k - \mathcal{G}_k(\mathbf{x}_0) \right]$$

- Now, $\mathbf{G}_k$ is the Jacobian of $\mathcal{G}_k$, and:

$$\begin{aligned} \mathcal{G}_k &= \mathcal{H}_k \circ \mathcal{M}_{t_0 \to t_k} \\ &= \mathcal{H}_k \circ \mathcal{M}_{t_{k-1} \to t_k} \circ \mathcal{M}_{t_{k-2} \to t_{k-1}} \cdots \mathcal{M}_{t_0 \to t_1} \end{aligned}$$

- Hence:

$$\mathbf{G}_k = \mathbf{H}_k \mathbf{M}_{t_{k-1} \to t_k} \mathbf{M}_{t_{k-2} \to t_{k-1}} \cdots \mathbf{M}_{t_0 \to t_1}$$

- And:

$$\mathbf{G}_k^{\mathrm{T}} = \mathbf{M}_{t_0 \to t_1}^{\mathrm{T}} \cdots \mathbf{M}_{t_{k-2} \to t_{k-1}}^{\mathrm{T}} \mathbf{M}_{t_{k-1} \to t_k}^{\mathrm{T}} \mathbf{H}_k^{\mathrm{T}}$$

Strong Constraint 4D-Var

- ✗ Let us consider how to evaluate the second term of $\nabla J(\mathbf{x}_0)$:

$$
\begin{aligned}
\sum_{k=0}^{K} \mathbf{G}_k^{\mathrm{T}} \mathbf{R}_k^{-1} \left[\mathbf{y}_k - \mathcal{G}_k(\mathbf{x}_0)\right] \\
= \; & \mathbf{H}_0^{\mathrm{T}} \mathbf{R}_0^{-1} \left[\mathbf{y}_0 - \mathcal{G}_0(\mathbf{x}_0)\right] \\
+ \; & \mathbf{M}_{t_0 \to t_1}^{\mathrm{T}} \mathbf{H}_1^{\mathrm{T}} \mathbf{R}_1^{-1} \left[\mathbf{y}_1 - \mathcal{G}_1(\mathbf{x}_0)\right] \\
+ \; & \mathbf{M}_{t_0 \to t_1}^{\mathrm{T}} \mathbf{M}_{t_1 \to t_2}^{\mathrm{T}} \mathbf{H}_2^{\mathrm{T}} \mathbf{R}_2^{-1} \left[\mathbf{y}_2 - \mathcal{G}_2(\mathbf{x}_0)\right] \\
& \vdots \\
+ \; & \mathbf{M}_{t_0 \to t_1}^{\mathrm{T}} \mathbf{M}_{t_1 \to t_2}^{\mathrm{T}} \cdots \mathbf{M}_{t_{K-1} \to t_K}^{\mathrm{T}} \mathbf{H}_K^{\mathrm{T}} \mathbf{R}_K^{-1} \left[\mathbf{y}_K - \mathcal{G}_K(\mathbf{x}_0)\right]
\end{aligned}
$$

Strong Constraint 4D-Var

- ✘ Let us consider how to evaluate the second term of $\nabla J(\mathbf{x}_0)$:

$$
\begin{aligned}
&\sum_{k=0}^{K} \mathbf{G}_k^{\mathrm{T}} \mathbf{R}_k^{-1} \left[\mathbf{y}_k - \mathcal{G}_k(\mathbf{x}_0) \right] \\
&= \mathbf{H}_0^{\mathrm{T}} \mathbf{R}_0^{-1} \left[\mathbf{y}_0 - \mathcal{G}_0(\mathbf{x}_0) \right] \\
&+ \mathbf{M}_{t_0 \to t_1}^{\mathrm{T}} \mathbf{H}_1^{\mathrm{T}} \mathbf{R}_1^{-1} \left[\mathbf{y}_1 - \mathcal{G}_1(\mathbf{x}_0) \right] \\
&+ \mathbf{M}_{t_0 \to t_1}^{\mathrm{T}} \mathbf{M}_{t_1 \to t_2}^{\mathrm{T}} \mathbf{H}_2^{\mathrm{T}} \mathbf{R}_2^{-1} \left[\mathbf{y}_2 - \mathcal{G}_2(\mathbf{x}_0) \right] \\
&\quad \vdots \\
&+ \mathbf{M}_{t_0 \to t_1}^{\mathrm{T}} \mathbf{M}_{t_1 \to t_2}^{\mathrm{T}} \cdots \mathbf{M}_{t_{K-1} \to t_K}^{\mathrm{T}} \mathbf{H}_K^{\mathrm{T}} \mathbf{R}_K^{-1} \left[\mathbf{y}_K - \mathcal{G}_K(\mathbf{x}_0) \right]
\end{aligned}
$$

Strong Constraint 4D-Var

- Let us consider how to evaluate the second term of $\nabla J(\mathbf{x}_0)$:

$$
\begin{aligned}
\sum_{k=0}^{K} \mathbf{G}_k^{\mathrm{T}} \mathbf{R}_k^{-1} \left[\mathbf{y}_k - \mathcal{G}_k(\mathbf{x}_0)\right]
&= \mathbf{H}_0^{\mathrm{T}} \mathbf{R}_0^{-1} \left[\mathbf{y}_0 - \mathcal{G}_0(\mathbf{x}_0)\right] \\
&+ \mathbf{M}_{t_0 \to t_1}^{\mathrm{T}} \mathbf{H}_1^{\mathrm{T}} \mathbf{R}_1^{-1} \left[\mathbf{y}_1 - \mathcal{G}_1(\mathbf{x}_0)\right] \\
&+ \mathbf{M}_{t_0 \to t_1}^{\mathrm{T}} \mathbf{M}_{t_1 \to t_2}^{\mathrm{T}} \mathbf{H}_2^{\mathrm{T}} \mathbf{R}_2^{-1} \left[\mathbf{y}_2 - \mathcal{G}_2(\mathbf{x}_0)\right] \\
&\vdots \\
&+ \mathbf{M}_{t_0 \to t_1}^{\mathrm{T}} \mathbf{M}_{t_1 \to t_2}^{\mathrm{T}} \cdots \mathbf{M}_{t_{K-1} \to t_K}^{\mathrm{T}} \mathbf{H}_K^{\mathrm{T}} \mathbf{R}_K^{-1} \left[\mathbf{y}_K - \mathcal{G}_K(\mathbf{x}_0)\right] \\
&= \mathbf{H}_0^{\mathrm{T}} \mathbf{R}_0^{-1} \left[\mathbf{y}_0 - \mathcal{G}_0(\mathbf{x}_0)\right] \\
&\quad + \mathbf{M}_{t_0 \to t_1}^{\mathrm{T}} \Big\{ \mathbf{H}_1^{\mathrm{T}} \mathbf{R}_1^{-1} \left[\mathbf{y}_1 - \mathcal{G}_1(\mathbf{x}_0)\right] \\
&\qquad + \mathbf{M}_{t_1 \to t_2}^{\mathrm{T}} \Big\{ \mathbf{H}_2^{\mathrm{T}} \mathbf{R}_2^{-1} \left[\mathbf{y}_2 - \mathcal{G}_2(\mathbf{x}_0)\right] \\
&\qquad\quad + \mathbf{M}_{t_2 \to t_3}^{\mathrm{T}} \Big\{ \cdots
\end{aligned}
$$

Strong Constraint 4D-Var

- Hence, to evaluate the gradient of the cost function, we can use the following algorithm:

  $\nabla J := 0;$

  for $k = K, K-1, \cdots, 1$ **do**

  ```
  /* Add the contribution of time t_k */
  ```
  $\nabla J := \nabla J - \mathbf{H}_k^{\mathrm{T}} \left[\mathbf{y}_k - \mathcal{G}_k(\mathbf{x}_k) \right];$

  ```
  /* Integrate the adjoint model from time t_k to time t_{k-1} */
  ```
  $\nabla J := \mathbf{M}_{t_{k-1} \rightarrow t_k}^{\mathrm{T}} \nabla J \;;$

  end

  ```
  /* Add the contribution from the observations at t_0, and
     the contribution from the background term: */
  ```
  $\nabla J := \nabla J - \mathbf{H}_0^{\mathrm{T}} \left[\mathbf{y}_0 - \mathcal{G}_0(\mathbf{x}_0) \right] + {\mathbf{P}^b}^{-1} (\mathbf{x}_0 - \mathbf{x}_b) \;;$

- Each $\mathbf{M}_{t_{k-1} \rightarrow t_k}^{\mathrm{T}}$ corresponds to a timestep of the adjoint model.
- Note that the adjoint model is integrated backwards in time, starting from t_K and ending with t_0.

Outline

The incremental Method

- We have seen how the 4D-Var cost function and gradient can be evaluated for the cost of
  - one integration of the forecast model
  - one integration of the adjoint model
- This cost is still prohibitive:
  - The cost of the adjoint model is typically 3 times that of the forward model.
  - The cost of one evaluation of the gradient is therefore about 4 times the forward model or about 2 days of forward model integration for a 12-hours assimilation window.
  - A typical minimisation will require between 10 and 100 evaluations of the gradient.
  - Hence, the cost of the analysis is roughly equivalent to between 20 and 200 days of model integration.
- The incremental algorithm was introduce to reduce the cost of 4D-Var by reducing the resolution of the model.

The incremental Method

- The incremental method can be applied to both 3D-Var and 4D-Var, so let's return to the general expression for the cost function:

$$J(\mathbf{x}) = \frac{1}{2}(\mathbf{x}-\mathbf{x}_b)^{\mathrm{T}}\mathbf{P}^{b^{-1}}(\mathbf{x}-\mathbf{x}_b) + \frac{1}{2}\left[\mathbf{y}-\mathcal{G}(\mathbf{x})\right]^{\mathrm{T}}\mathbf{R}^{-1}\left[\mathbf{y}-\mathcal{G}(\mathbf{x})\right]$$

- We introduce a linearisation state $\mathbf{x}^{(m)}$, for $m \in \{1, 2, \cdots, M\}$, and write

$$\mathbf{x} = \mathbf{x}^{(m)} + \delta\mathbf{x}^{(m)}$$

- We linearise the generalised observation operators around $\mathbf{x}^{(m)}$:

$$\begin{aligned} \mathcal{G}(\mathbf{x}) &= \mathcal{G}\left(\mathbf{x}^{(m)} + \delta\mathbf{x}^{(m)}\right) \\ &\approx \mathcal{G}\left(\mathbf{x}^{(m)}\right) + \mathbf{G}\delta\mathbf{x}^{(m)} \end{aligned}$$

- We introduce the innovation $\mathbf{d}^{(m)}$:

$$\mathbf{d}^{(m)} = \mathbf{y} - \mathcal{G}\left(\mathbf{x}^{(m)}\right),$$

- and the background increment $\delta\mathbf{x}_b$ as:

$$\mathbf{x}_b = \mathbf{x}^{(m)} + \delta\mathbf{x}_b$$

The incremental Method

- The cost function can be written in terms of the increment $\delta\mathbf{x}^{(m)}$, and approximated by the quadratic function:

$$\begin{aligned} J\left(\delta\mathbf{x}^{(m)}\right) &= \frac{1}{2}\left[\delta\mathbf{x}^{(m)} - \delta\mathbf{x}_b\right]^{\mathrm{T}} \mathbf{P}^{b^{-1}} \left[\delta\mathbf{x}^{(m)} - \delta\mathbf{x}_b\right] \\ &+ \frac{1}{2}\left[\mathbf{d}^{(m)} - \mathbf{G}\delta\mathbf{x}^{(m)}\right]^{\mathrm{T}} \mathbf{R}^{-1} \left[\mathbf{d}^{(m)} - \mathbf{G}\delta\mathbf{x}^{(m)}\right] \end{aligned}$$

- The incremental method treats the minimisation of J as a sequence of quadratic problems:

for $m = 0, 1, \cdots, M$ **do**

`/* Minimise the quadratic cost function` $J\left(\delta\mathbf{x}^{(m)}\right)$ `*/`

$\delta\mathbf{x}_a^{(m)} = \arg\min_{\mathbf{x}}\left[J\left(\delta\mathbf{x}^{(m)}\right)\right]$;

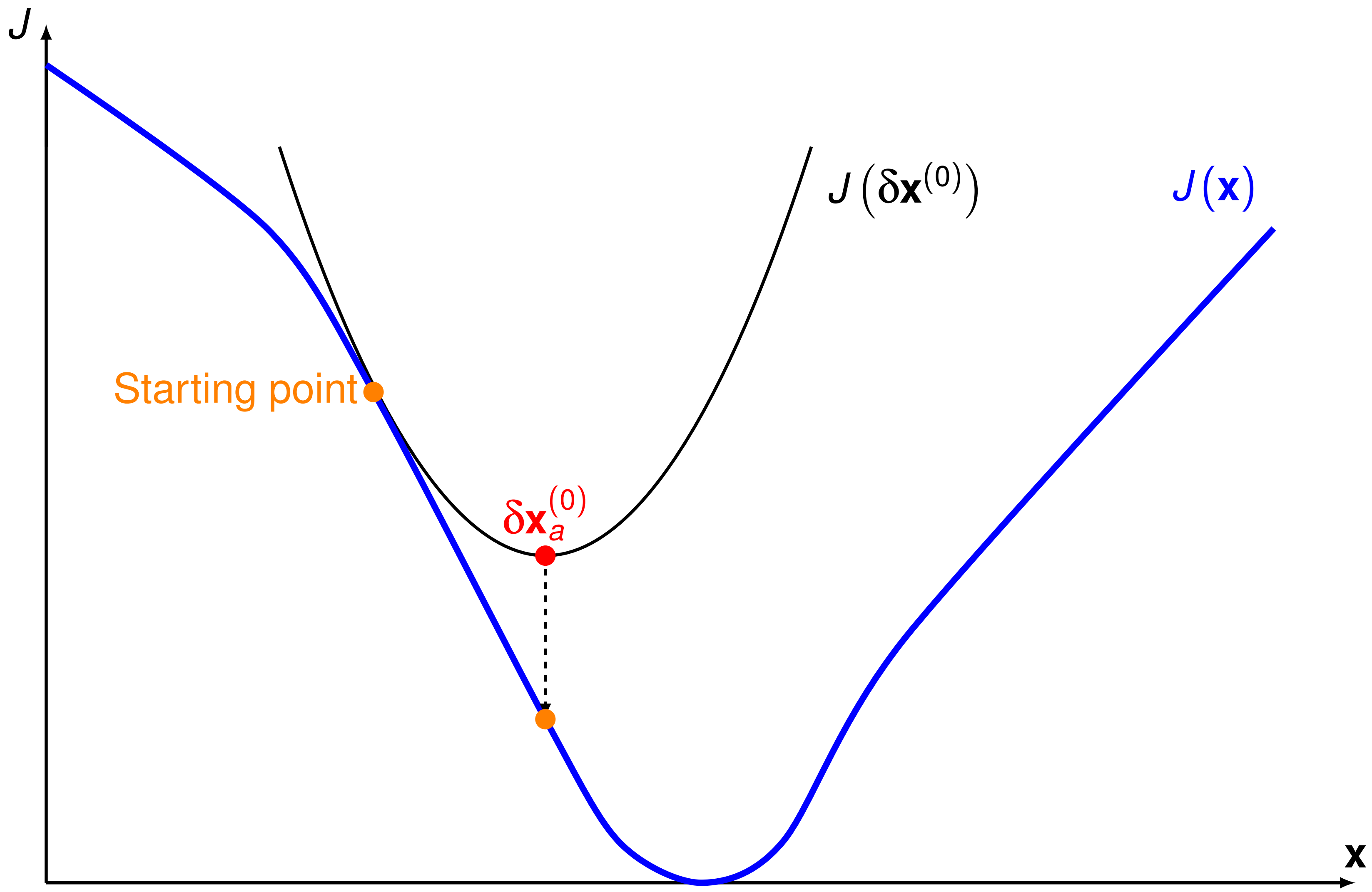
`/* Set the new linearisation state */`

$\mathbf{x}^{(m+1)} = \mathbf{x}^{(m)} + \delta\mathbf{x}_a^{(m)}$;

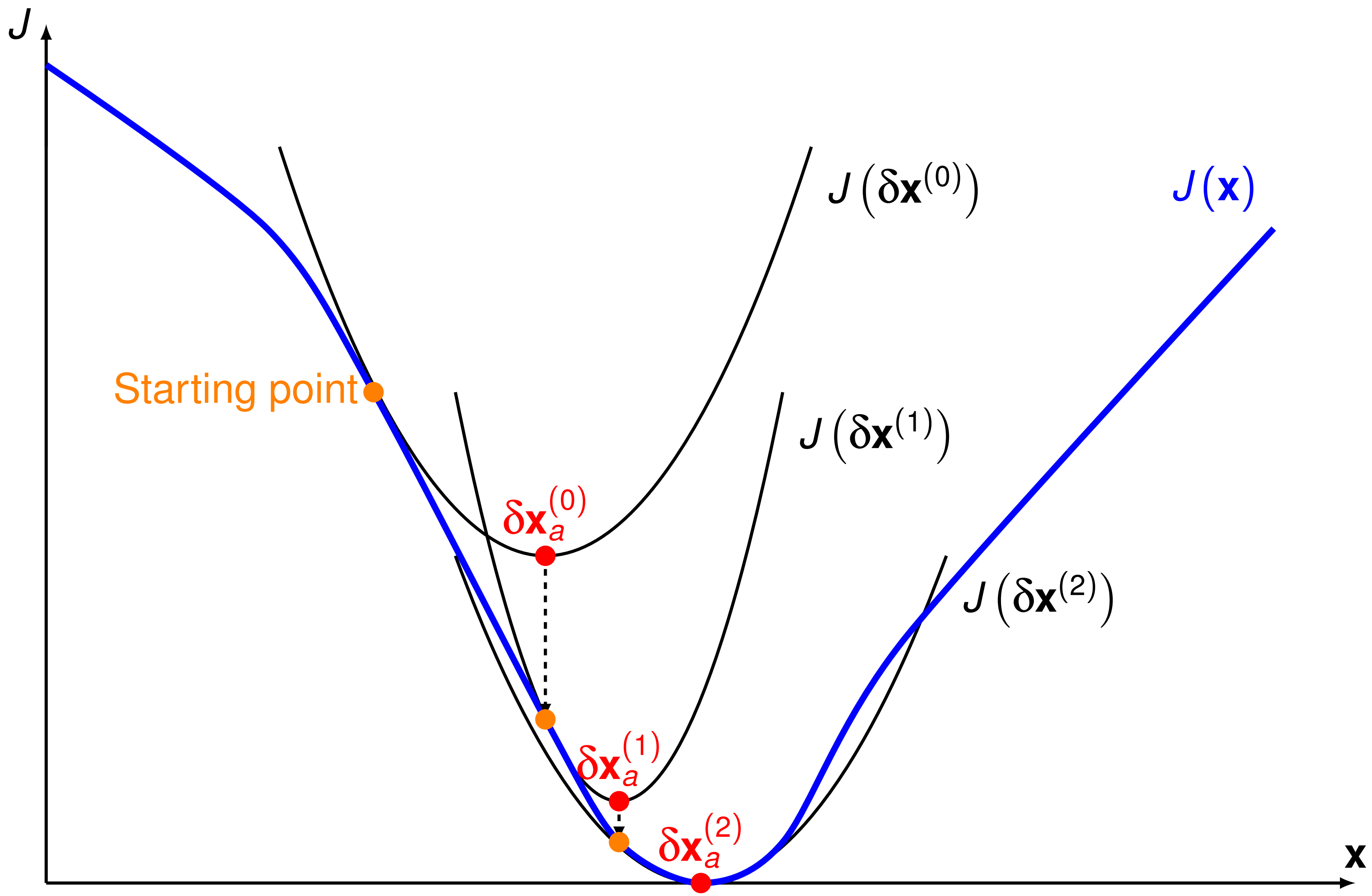
end

- In this form, if the minimisation converges, it will converge to the solution of the original problem.

The incremental Method



The incremental Method



The incremental method with Lanczos

for $m = 0, 1, \cdots, M$ **do**

```
/* Change of variable                                                   */
```
$\chi^{(m)} = \mathbf{L}^{-1}\left(\mathbf{x}^{(m)} - \mathbf{x}^{b}\right);$

```
/*                                                                      */
/* Minimise the quadratic cost function J(χ^(m))                        */
/* If m > 0 precondition with I + Σ_{i=1}^K (λ_i^{-1/2} - 1) v_i v_i^T ≈ (J'')^{-1/2} */
/*                                                                      */
```
$\chi_a^{(m)} = \arg\min_{\chi}\left[J\left(\chi^{(m)}\right)\right];$

```
/*                                                                      */
/* Compute the first K eigenvalues/vectors of J''                       */
```
$\lambda_i = \cdots;$

$\mathbf{v}_i = \cdots;$

```
/*                                                                      */
/* Set the new linearisation state                                      */
```
$\mathbf{x}^{(m+1)} = \mathbf{x}^{(m)} + \mathbf{L}\chi_a^{(m)};$

```
/*                                                                      */
```

end

The incremental Method

- However, to reduce the computational cost of the analysis, we can make a further approximation, and evaluate the quadratic cost function at lower resolution:

$$
\begin{aligned}
J\left(\delta\widetilde{\mathbf{x}}^{(m)}\right) &= \frac{1}{2}\left(\delta\widetilde{\mathbf{x}}^{(m)} - \delta\widetilde{\mathbf{x}}_b\right)^{\mathrm{T}} \widetilde{\mathbf{P}}^{b\,-1}\left(\delta\widetilde{\mathbf{x}}^{(m)} - \delta\widetilde{\mathbf{x}}_b\right) \\
&+ \frac{1}{2}\left[\mathbf{d}^{(m)} - \widetilde{\mathbf{G}}\delta\widetilde{\mathbf{x}}^{(m)}\right]^{\mathrm{T}} \mathbf{R}^{-1}\left[\mathbf{d}^{(m)} - \widetilde{\mathbf{G}}\delta\widetilde{\mathbf{x}}^{(m)}\right]
\end{aligned}
$$

  where $\tilde{\cdot}$ indicates low resolution, and where $\widetilde{\mathbf{x}}_b$, etc. are interpolated from the corresponding full-resolution fields.

- When the quadratic cost function is approximated in this way, 4D-Var no longer converges to the solution of the original problem.

The Incremental Method

- Incremental formulation:

$$J\left(\delta\widetilde{\mathbf{x}}^{(m)}\right) = \frac{1}{2}\left(\delta\widetilde{\mathbf{x}}^{(m)} - \delta\widetilde{\mathbf{x}}_b\right)^{\mathrm{T}} \widetilde{\mathbf{P}}^{b\,-1}\left(\delta\widetilde{\mathbf{x}}^{(m)} - \delta\widetilde{\mathbf{x}}_b\right) + \frac{1}{2}\left[\mathbf{d}^{(m)} - \widetilde{\mathbf{G}}\delta\widetilde{\mathbf{x}}^{(m)}\right]^{\mathrm{T}} \mathbf{R}^{-1}\left[\mathbf{d}^{(m)} - \widetilde{\mathbf{G}}\delta\widetilde{\mathbf{x}}^{(m)}\right]$$

- The analysis increments are calculated at reduced resolution and must be interpolated to the high-resolution model's grid.

- Note, however that $\mathbf{d}^{(m)} = \mathbf{y} - \mathcal{G}\left(\mathbf{x}^{(m)}\right)$ is evaluated using the full-resolution versions of $\mathcal{G}$ and $\mathbf{x}^{(m)}$.

- I.e. the observations are always compared with the *full resolution* linearisation state. The reduced-resolution observation operator only appears applied to increments: $\widetilde{\mathbf{G}}\delta\widetilde{\mathbf{x}}^{(m)}$.

Outline

FGAT formulation

- The first guess at appropriate time (FGAT) formulation was introduced to further reduce the cost of the 4D-Var.
- Cost function in terms of the increment $\delta\mathbf{x}_0$:

$$J(\delta\mathbf{x}_0) = \frac{1}{2}[\delta\mathbf{x}_0 - \delta\mathbf{x}_b]^{\mathrm{T}} \mathbf{P}^{b^{-1}} [\delta\mathbf{x}_0 - \delta\mathbf{x}_b] + \frac{1}{2}\sum_{k=0}^{K}[\mathbf{d}_k - \mathbf{G}_k\delta\mathbf{x}_0]^{\mathrm{T}} \mathbf{R}_k^{-1} [\mathbf{d}_k - \mathbf{G}_k\delta\mathbf{x}_0] \, ,$$

  with:
  - $\mathbf{d}_k = \mathbf{y}_k - \mathcal{G}_k(\mathbf{x}_0)$
  - $\mathbf{G}_k = \mathbf{H}_k\mathbf{M}_{t_{k-1}\to t_k}\mathbf{M}_{t_{k-2}\to t_{k-1}}\cdots\mathbf{M}_{t_0\to t_1}$
- FGAT approximation:
  - $\mathbf{M}_{t_{k-1}\to t_k} \equiv \mathbf{I}$ for $k = 1, 2, \cdots, K$.

$$J(\delta\mathbf{x}_0) = \frac{1}{2}[\delta\mathbf{x}_0 - \delta\mathbf{x}_b]^{\mathrm{T}} \mathbf{P}^{b^{-1}} [\delta\mathbf{x}_0 - \delta\mathbf{x}_b] + \frac{1}{2}\sum_{k=0}^{K}[\mathbf{d}_k - \mathbf{H}_k\delta\mathbf{x}_0]^{\mathrm{T}} \mathbf{R}_k^{-1} [\mathbf{d}_k - \mathbf{H}_k\delta\mathbf{x}_0] \, .$$

FGAT formulation

$$
\begin{aligned}
J(\delta\mathbf{x}_0) &= \frac{1}{2}[\delta\mathbf{x}_0 - \delta\mathbf{x}_b]^{\mathrm{T}}\,\mathbf{P}^{b^{-1}}\,[\delta\mathbf{x}_0 - \delta\mathbf{x}_b] \\
&+ \frac{1}{2}\sum_{k=0}^{K}[\mathbf{d}_k - \mathbf{H}_k\delta\mathbf{x}_0]^{\mathrm{T}}\,\mathbf{R}^{-1}\,[\mathbf{d}_k - \mathbf{H}_k\delta\mathbf{x}_0] \\
\nabla J(\delta\mathbf{x}_0) &= \mathbf{P}^{b^{-1}}\,[\delta\mathbf{x}_0 - \delta\mathbf{x}_b] - \sum_{k=0}^{K}\mathbf{H}_k^{\mathrm{T}}\mathbf{R}_k^{-1}\,[\mathbf{d}_k - \mathbf{H}_k\delta\mathbf{x}_0]\ .
\end{aligned}
$$

- ✘ No model integration in the minimisation:
  - ➩ Computationally much cheaper than general 4D-Var.
  - ➩ Yet the comparison between the model and the observations is computed at the right observation time.
- ✘ Analysis increment valid for the entire analysis window:
  - ➩ The model should vary slowly within the analysis window preferably. This is the case for the ocean when the analysis window is 24 hr (see lecture on Friday).
  - ➩ Analysis increment usually added in the middle of the analysis window.
- ✘ Could be combined with incremental approach.

Outline

Weak Constraint 4D-Var

- ✘ The perfect model assumption limits the length of analysis window that can be used to roughly 12 hours (for an NWP system).
- ✘ To use longer analysis windows (or to account for deficiencies of the model that are already apparent with a 12-hour window) we must relax the perfect model assumption.
- ✘ We saw already that strong constraint 4D-Var can be expressed as:

$$\begin{aligned} \mathbf{x}_a &= \arg\min_{\mathbf{x}_0} \left(J(\mathbf{x}_0, \mathbf{x}_1, \cdots \mathbf{x}_k) \right) \\ \text{subject to} \quad \mathbf{x}_k &= \mathcal{M}_{t_{k-1} \rightarrow t_k} \left(\mathbf{x}_{k-1} \right) \quad \text{for } k = 1, 2, \cdots, K \end{aligned}$$

- ✘ In weak constraint 4D-Var, we define the model error η_k as

$$\mathbf{x}_k = \mathcal{M}_{t_{k-1} \rightarrow t_k} \left(\mathbf{x}_{k-1} \right) + \eta_k \quad \text{for } k = 1, 2, \cdots, K$$

  and we allow η_k to be non-zero.
- ✘ Note that now $\mathbf{x}_0$ is not enough to determine the states $\mathbf{x}_k$ for $k = 1, \cdots, K$

Weak Constraint 4D-Var

- We can derive the weak constraint cost function using Bayes' rule:

$$p(\mathbf{x}_0 \cdots \mathbf{x}_K | \mathbf{y}_0 \cdots \mathbf{y}_K) = \frac{p(\mathbf{y}_0 \cdots \mathbf{y}_K | \mathbf{x}_0 \cdots \mathbf{x}_K) p(\mathbf{x}_0 \cdots \mathbf{x}_K)}{p(\mathbf{y}_0 \cdots \mathbf{y}_K)}$$

- The denominator is independent of $\mathbf{x}_0 \cdots \mathbf{x}_K$.
- The first term of the numerator simplifies to:

$$p(\mathbf{y}_0 \cdots \mathbf{y}_K | \mathbf{x}_0 \cdots \mathbf{x}_K) = \prod_{k=0}^{K} p(\mathbf{y}_k | \mathbf{x}_k) \,,$$

assuming that the observation $\mathbf{y}_k$ is independent of $\mathbf{x}_l$ for $l \neq k$.

- Hence

$$p(\mathbf{x}_0 \cdots \mathbf{x}_K | \mathbf{y}_0 \cdots \mathbf{y}_K) \propto \left[\prod_{k=0}^{K} p(\mathbf{y}_k | \mathbf{x}_k) \right] p(\mathbf{x}_0 \cdots \mathbf{x}_K)$$

Weak Constraint 4D-Var

$$p(\mathbf{x}_0 \cdots \mathbf{x}_K | \mathbf{y}_0 \cdots \mathbf{y}_K) \propto \left[\prod_{k=0}^{K} p(\mathbf{y}_k | \mathbf{x}_k) \right] p(\mathbf{x}_0 \cdots \mathbf{x}_K)$$

- Taking minus the logarithm gives the cost function:

$$J(\mathbf{x}_0 \cdots \mathbf{x}_K) = - \sum_{k=0}^{K} \log \left[p(\mathbf{y}_k | \mathbf{x}_k) \right] - \log \left[p(\mathbf{x}_0 \cdots \mathbf{x}_K) \right]$$

- The term involving $\mathbf{y}_k$ is familiar. It is the observation term of the strong constraint cost function.
- The final term is less familiar. It represents the *a priori* probability of the sequence of states $\mathbf{x}_0 \cdots \mathbf{x}_K$.

Weak Constraint 4D-Var

- Given the sequence of states $\mathbf{x}_0 \cdots \mathbf{x}_K$, we can calculate the corresponding model errors:

$$\eta_k = \mathbf{x}_k - \mathcal{M}_{t_{k-1} \rightarrow t_k}(\mathbf{x}_{k-1}) \quad \text{for } k = 1, 2, \cdots, K$$

- We can use our knowledge of the statistics of model error to define

$$p(\mathbf{x}_0 \cdots \mathbf{x}_K) \equiv p(\mathbf{x}_0; \eta_1 \cdots \eta_K)$$

- One significant assumption is to assume that model error is uncorrelated in time. In this case:

$$p(\mathbf{x}_0 \cdots \mathbf{x}_K) \equiv p(\mathbf{x}_0) \prod_{k=1}^{K} p(\eta_k)$$

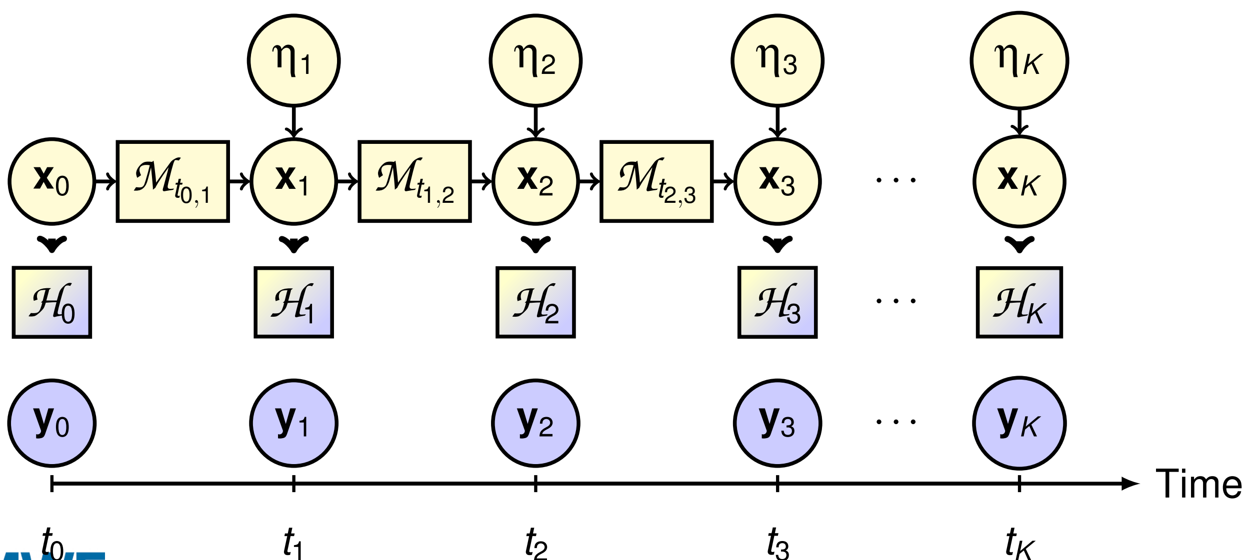
- $p(\mathbf{x}_0)$ is familiar. It is the background term of the strong constraint cost function.
- If we take $p(\eta_k)$ as Gaussian with covariance matrix $\mathbf{Q}_k$, we see that weak constraint 4D-Var adds the following term to the cost function:

$$\frac{1}{2} \sum_{K=1}^{K} \eta_k^{\mathrm{T}} \mathbf{Q}_k^{-1} \eta_k$$

Weak Constraint 4D-Var

- Hence, for Gaussian, temporally-uncorrelated model error, the weak constraint cost function is:

$$
\begin{aligned}
J(\mathbf{x}_0, \mathbf{x}_1, \cdots \mathbf{x}_k) &= \frac{1}{2}(\mathbf{x}_0 - \mathbf{x}_b)^{\mathrm{T}} \mathbf{P}^{b^{-1}} (\mathbf{x}_0 - \mathbf{x}_b) \\
&+ \frac{1}{2} \sum_{k=0}^{K} \left[\mathbf{y}_k - \mathcal{H}_k(\mathbf{x}_k)\right]^{\mathrm{T}} \mathbf{R}_k^{-1} \left[\mathbf{y}_k - \mathcal{H}_k(\mathbf{x}_k)\right] \\
&+ \frac{1}{2} \sum_{k=1}^{K} \eta_k^{\mathrm{T}} \mathbf{Q}_k^{-1} \eta_k \quad \text{where} \quad \mathbf{x}_k = \mathcal{M}_{t_{k-1} \to t_k}(\mathbf{x}_{k-1}) + \eta_k .
\end{aligned}
$$



Weak Constraint 4D-Var

- In strong constraint 4D-Var, we can use the constraints to reduce the problem of minimising a function of $\mathbf{x}_0 \cdots \mathbf{x}_K$ to that of minimising a function of the initial state $\mathbf{x}_0$ only.
- This is not possible in weak constraint 4D-Var — we must either:
  - minimise the function $J(\mathbf{x}_0 \cdots \mathbf{x}_K)$, or:
  - express the cost function as a function of $\mathbf{x}_0$ and $\eta_1 \cdots \eta_K$.
- Although the two approaches are mathematically equivalent, they lead to very different minimisation problems, with different possibilities for preconditioning.
  - It is not yet clear which approach is the best.
  - Formulation of an incremental method for weak constraint 4D-Var also remains a topic of research.
- Finally, note that model error is unlikely to be temporally uncorrelated.
  - Indeed, initial attempts to account for model error in the ECMWF analysis are concentrated on representing only the bias component of model error (i.e. model error is assumed constant over the analysis window).
- There is a whole 1-hour lecture on model error Thursday.

Outline

Summary

- Strong Constraint 4D-Var is an extension of 3D-Var to the case where observations are distributed in time.
- The observation operators are generalised to include an integration of the forecast model.
- The model is assumed to be perfect, so that the four-dimensional analysis state corresponds to an integration (trajectory) of the model.
- The incremental method allows the computational cost to be reduced to acceptable levels.
- The FGAT approximation allows to further reduce the cost but relies on strong assumption on the model evolution.
- Weak Constraint 4D-Var allows the perfect model assumption to be removed.