TRANSFORMS AND PRECONDITIONING IN
THE MET OFFICE 3D VAR SCHEME
Ross Bannister
D.A.R.C.

Cost Function in \( w \)-space (reduced resolution, real space):
\[
J (\hat{w}') = J_b (\hat{w}') + J_o (\hat{w}')
\]
\[
J_b (\hat{w}') = \frac{1}{2} (\hat{w}^b' - \hat{w}')^T B^{-1} (\hat{w}^b' - \hat{w}')
\]
\[
J_o (\hat{w}') = \frac{1}{2} (\hat{y}^o - H (\hat{w}', \hat{w}^o))^T (E + F)^{-1} (\hat{y}^o - H (\hat{w}', \hat{w}^o))
\]

Gradients w.r.t. \( w' \):
\[
\frac{dJ_b}{d\hat{w}'} = -B^{-1} (\hat{w}^b' - \hat{w}')
\]
\[
\frac{dJ_o}{d\hat{w}'} = -H^T (E + F)^{-1} (\hat{y}^o - H (\hat{w}', \hat{w}^o))
\]

Hessian,
\[
\frac{d^2J_b}{d\hat{w}'^2} = B^{-1}
\]
\[
\frac{d^2J_o}{d\hat{w}'^2} = H^T (E + F)^{-1} H
\]

Contruct a control variable, \( \tilde{v} \):
Transform
\[
\tilde{w}' = U (\tilde{v})
\]
\[
\frac{d^2J_b}{d\tilde{v}^2} = I
\]
Inverse
\[
\tilde{v} = T (\tilde{w}')
\]
Such that
What is the basis of a $\tilde{w}$ vector?

Perturbations are from a guess state:

$$\tilde{w} = \tilde{w} - \tilde{w}_g$$

$$\tilde{w}^{rb} = \tilde{w}^b - \tilde{w}_g$$

Why do we want to make a transformation to $\tilde{v}$-space?

- Makes the problem manageable.
- Preconditioning - makes the minimization process more efficient and accurate.
- No preconditioning: (Largest e.v.)/(smallest e.v.) $\sim 10^{10}$ !
- Makes the scheme more complicated to understand.
- Balance problems?
What is the principle of the preconditioning transform?
To make the 'weight' of each control variable equal.

\[ \tilde{w}' = U\tilde{v} \quad \tilde{v} = Tw' \]

\[ U = T^{-1} \]

\[ J_b = \frac{1}{2} (U\tilde{v}^b - U\tilde{v})^T B^{-1} (U\tilde{v}^b - U\tilde{v}) \]

\[ = \frac{1}{2} (\tilde{v}^b - \tilde{v})^T U^T B^{-1} U (\tilde{v}^b - \tilde{v}) \]

Choose \( U \) such that \( U^T B^{-1} U = I \)

\[ U^{-1} B (U^T)^{-1} = I \quad \Rightarrow \quad B = UU^T \]

\[ TBT^T = I \]

\( U \) is not a unitary or orthogonal transform, instead it is like the square-root of \( B \).

The information regarding the covariances is transferred into the transformation itself (and inverts \( B \! \)).
This is done in two steps:

\[
\text{Let } U = U_2 U_1
\]

\[
U_1^{-1} U_2^{-1} B (U_2^T)^{-1} (U_1^T)^{-1} = I \quad \ast
\]

Consider \( B \) afresh. Diagonalize with a transform \( Y^T \):

\[
Y^T B Y = \Lambda
\]

eigenfunctions, rows of \( Y^T \)

eigenvalues, diagonal matrix \( \Lambda \)

There are, by definition, no co-variances between the eigenmodes. Can now 'remove' the variance by:

\[
\Lambda^{-1/2} Y^T B Y \Lambda^{-1/2} = I
\]

c.f. (\( \ast \)) to show that:

\[
U_1 = \Lambda^{1/2} \quad \Rightarrow \quad T_1 = \Lambda^{-1/2}
\]

\[
U_2 = Y \quad \Rightarrow \quad T_2 = Y^T
\]

Problem: \( B \) is too large to work with (even at half resolution).

<table>
<thead>
<tr>
<th># fields</th>
<th># long. points</th>
<th># lat. points</th>
<th># levels</th>
<th># elements ( B )</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>216</td>
<td>163</td>
<td>30</td>
<td>&gt; 10^{13}</td>
</tr>
<tr>
<td>5</td>
<td>48</td>
<td>37</td>
<td>42</td>
<td>&gt; 10^{11}</td>
</tr>
</tbody>
</table>
The solution in three easy stages ...

Assume that:

- We can choose an alternative set of physical parameters which are only weakly correlated,
- The covariances within each parameter can be 'removed' separately (e.g. vertical and horizontal parts normalized independently).
- We can use the last section as a guide.

(i) The first stage of the T-transform (parameter transform).
(ii) The second stage is a vertical transformation.
(iii) The third stage is a horizontal transformation.

\[ T = T_h T_v T_p \]

(i) The parameter transform

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Eqs.</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \psi )</td>
<td>( \nabla^2 \psi = \vec{k} \cdot \nabla \times \vec{u} )</td>
</tr>
<tr>
<td>( \chi )</td>
<td>( \nabla^2 \chi = \nabla \cdot \vec{u} )</td>
</tr>
<tr>
<td>( A_p )</td>
<td>( p = Gp + Ap )</td>
</tr>
<tr>
<td>( \mu )</td>
<td>( q / q_{\text{sat}} )</td>
</tr>
</tbody>
</table>

\[
\begin{array}{cccc}
\psi & \chi & A_p & \mu \\
B_\psi & 0 & 0 & 0 \\
0 & B_\chi & 0 & 0 \\
0 & 0 & B_{A_p} & 0 \\
0 & 0 & 0 & B_\mu \\
\end{array}
\]

\( B^{-1} \) has a similar structure
Background term is written (in terms of parameter perturbations),

\[
J_B(\nu_p) = \frac{1}{2} \left( \bar{\nu}_p^b - \tilde{\nu}_p \right)^T B^{-1} \left( \bar{\nu}_p^b - \tilde{\nu}_p \right)
\]

\[
= \frac{1}{2} \left( \bar{\psi}^b - \bar{\psi} \right)^T B_{\psi}^{-1} \left( \bar{\psi}^b - \bar{\psi} \right) + \frac{1}{2} \left( \bar{\chi}^b - \bar{\chi} \right)^T B_{\chi}^{-1} \left( \bar{\chi}^b - \bar{\chi} \right) + \ldots
\]

\[
\tilde{\nu}_p = T_p \tilde{\nu}' = \begin{pmatrix} \bar{\psi} \\ \bar{\chi} \\ \bar{\mu} \end{pmatrix}
\]

Cov. matrices for each parameter (e.g. \(\psi\)) - outer or tensor product:

\[
\text{Cov} = (\bar{\psi} - \bar{\psi}_i) (\bar{\psi} - \bar{\psi}_i)^T
\]

The size of these covariance matrices is still too large.

Do remaining transformations for each parameter separately.
(ii) Vertical transform

<table>
<thead>
<tr>
<th>Aim (each parameter):</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reformulate (i) the state variable and (ii) the cov. matrix in</td>
</tr>
<tr>
<td>terms of modes which are uncorrelated in the vertical, each of</td>
</tr>
<tr>
<td>which having unit variance.</td>
</tr>
</tbody>
</table>

We can calculate the vertical covariances for each column:

\[
\text{Cov}(\lambda, \phi; \ell, \ell') = \frac{(\psi(\lambda, \phi, t; \ell) - \psi_i(\lambda, \phi, t; \ell)) \times (\psi(\lambda, \phi, t; \ell') - \psi_i(\lambda, \phi, t; \ell'))}{(\psi(\lambda, \phi, t; \ell) - \psi_i(\lambda, \phi, t; \ell'))}
\]

Average over all \((\lambda, \phi)\) to form global covariance matrix:

\[\text{Let } B^\text{vert}_\psi(\ell, \ell') = \langle \text{Cov}(\lambda, \phi; \ell, \ell') \rangle\]

Decompose this such that \(B^\text{vert}_\psi = I\), as before (with weighting).

\[
\Lambda_v^{-1/2}F_v^T P B^\text{vert}_\psi P F_v \Lambda_v^{-1/2} = I
\]

\[
T^\text{vert}_v \text{ (global av.) } = \Lambda_v^{-1/2}F_v^T P
\]

How can we make this into a transform acting on \(\hat{\psi}\)?
Transforming the state vector ($\tilde{\psi}$):

$$\tilde{\psi}_{EOF} = T_v \psi$$

Transforming the error covariance matrix:

$$T_v B_\psi T_v^T$$

Include lat. variation

$$\Lambda_v \rightarrow \Lambda_v(\phi)$$

$$\Lambda_v(\phi) = F_v^T P B_\psi^{ver}(\phi) P F_v$$
Transforming the state vector: \( \vec{\nu}_v = T_v \vec{\nu}_v \)

Instead of labelling the vertical co-ordinate with 'level', label with 'EOF index'.

Quantities are perturbations

Think of surfaces of constant vertical EOF index.

This is the result of the vertical transform.

Horizontal transform
(iii) Horizontal Transform

Aim (each parameter):
Reformulate (i) the state variable and (ii) the cov. matrix in terms of modes which are \textbf{uncorrelated} in the horizontal, each of which having \textbf{unit variance}.

Decompose into modes which we assume are uncorrelated. Effectively (for one $\psi$-EOF surface):

$$\Lambda_h^{-1/2} F_h^T P B^\text{hor}_\psi P F_h \Lambda_h^{-1/2} = I$$

$$T^\text{hor}_h = \Lambda_h^{-1/2} F_h^T P$$

$B^\text{hor}_\psi$ is not explicitly calculated. Choose:

$P$ as a weight matrix, different from before,
$F_v^T$ as a horizontal spectral transform, and
$\Lambda_h^{1/2}$ as the correlation spectrum of the modes.

What does the horizontal transform look like (acts on $\hat{\psi}_{EOF}$)?
Transforming the state vector ($\hat{\psi}$):

$$\hat{\psi}_{v-space} = T_h \hat{\psi}_{EOF}$$

$T_h^{hor}$ Transform associated with surface EOF1

$T_h^{hor}$ Transform associated with surface EOF2

$T_h^{hor}$ Transform associated with surface EOF3, etc

Transforming the error covariance matrix:

$$T_h T_{\psi} B_{\psi} T_{\psi}^T T_h^T \approx I$$
The transformed state vector: \( \tilde{\mathbf{v}} = T_h \hat{\mathbf{v}} \)

Instead of labelling the horizontal co-ordinates with 'long/lat', label with 'HM index'

Quantities are perturbations

This is the result of all three transforms

Perform descent algorithm in this space
Summary of Equations

1. $\tilde{w}$-space formulation

   \[ J (\tilde{w}') = \frac{1}{2} (\tilde{w}'^b - \tilde{w}')^T B^{-1} (\tilde{w}'^b - \tilde{w}') + \]

   \[ \frac{1}{2} (\tilde{y}^o - H (\tilde{w}', \tilde{w}^g)) ^T (E + F)^{-1} (\tilde{y}^o - H (\tilde{w}', \tilde{w}^g)) \]

   \[ \frac{dJ}{d\tilde{w}'} = -B^{-1} (\tilde{w}'^b - \tilde{w}') - H ^T (E + F)^{-1} (\tilde{y}^o - H (\tilde{w}', \tilde{w}^g)) \]

   \[ \frac{d^2 J_b}{d\tilde{w}'^2} = B^{-1} + H ^T (E + F)^{-1} H \]

   $\tilde{w}$ = model state (’ pertbtn, $b$ backgrnd, $g$ guess)

   \[ H (\tilde{w}', \tilde{w}^g) \approx H (\tilde{w}^g) + H \tilde{w}' \]

2. $\tilde{v}$-space formulation

   \[ \tilde{v} = T \tilde{w}' \quad \tilde{w}' = U \tilde{v} \quad U^T B^{-1} U = I \]

   \[ J (\tilde{v}) = \frac{1}{2} (\tilde{v}'^b - \tilde{v}')^T U^T B^{-1} U (\tilde{v}'^b - \tilde{v}') + \]

   \[ \frac{1}{2} (\tilde{y}^o - H (U \tilde{v}, \tilde{w}^g)) ^T (E + F)^{-1} (\tilde{y}^o - H (U \tilde{v}, \tilde{w}^g)) \]

   \[ \frac{dJ}{d\tilde{v}} = -U^T B^{-1} U (\tilde{v}'^b - \tilde{v}) - U^T H ^T (E + F)^{-1} (\tilde{y}^o - H (U \tilde{v}, \tilde{w}^g)) \]

   \[ \frac{d^2 J_b}{d\tilde{v}^2} = U^T B^{-1} U + U^T H ^T (E + F)^{-1} H U \]