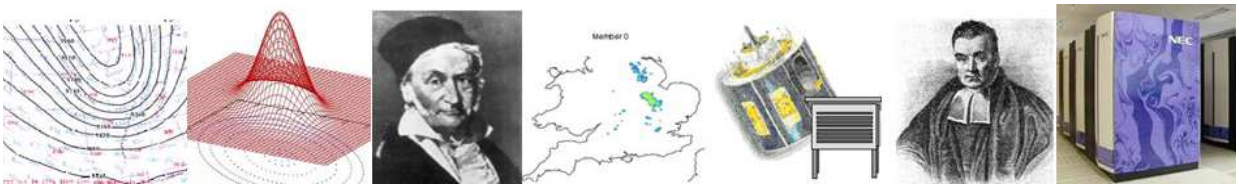
MSc Course: Theory and Techniques of Data Assimilation

Two Lectures on "3d-Var."



Ross Bannister, Room 1U11, Dept. of Meteorology, Univ. of Reading
r.n.bannister@reading.ac.uk
Version 2010

Section A: List Of Topics And References

A.1: List Of Topics

A. References.

B. Introduction - why do data assimilation?

C. 3-dimensional variational assimilation and operational data assimilation.

D. The gradient and Hessian of the cost function.

E. Example observation operators.

F. Minimization algorithms.

G. Preconditioning.

A.2: Further Reading

- Kalnay E., Atmospheric Modelling, *Data Assimilation and Predictability*, Ch. 5.
- Daley, *Atmospheric Data Analysis*, Ch.13.
- ECMWF, Data assimilation course handouts, `http://www.ecmwf.int/newsevents/training/lecture_notes/LN_DA.html`.
- Schlatter T.W., *Variational assimilation of meteorological observations in the lower atmosphere: a tutorial on how it works*, Journal of atmospheric and solar-terrestrial physics 62, pp. 1057-1070 (2000).
- Lorenc et al., *The Met Office global 3-dimensional variational assimilation scheme*, QJRMS 126, pp. 2991-3012 (2000).

Section B: The Need To Do Data Assimilation

B.1: Why do we need to do data assimilation (DA)?

DA is a tool that combines obs. and models and is used to infer knowledge about a dynamical system (the atmosphere, the oceans, etc.).

- DA can estimate the initial conditions (called the 'analysis') of weather or ocean forecast models. An atmospheric analysis may include fields of wind, temperature, pressure, humidity, and concentration of trace gases like ozone.
- DA has other applications, e.g. to generate scientific datasets at regular time intervals.
- Obs. and model data each used on their own have inadequacies. Used together with DA, their advantages can be combined (see table).
- DA can also estimate the uncertainty in the analysis.
- Bjerknes, 1911: The "*ultimate problem in meteorology*".
- Leith, 1993: The atmosphere "*is a chaotic system in which errors introduced into the system can grow with time ... As a consequence, data assimilation is a struggle between chaotic destruction of knowledge and its restoration by new observations*" (Fig. 1a).

	Pros	Cons
Observations	'Close' to reality.	Irregular and incomplete coverage (Fig 1b). May not be a direct measurement. Observations have errors.
Modelled data	Complete global coverage. Can be processed (e.g. differentiated).	Temporal growth of error in initial conditions (Fig 1a). Susceptibility to significant model error.

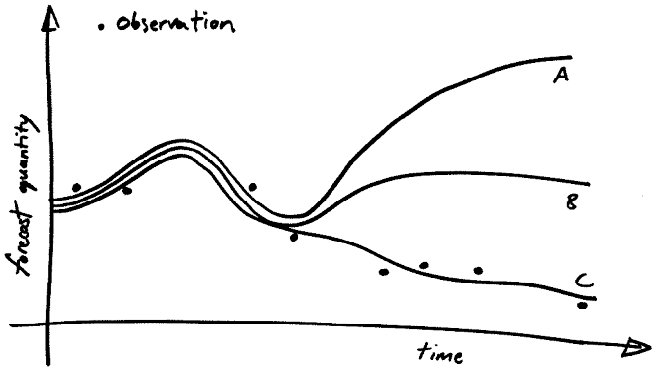
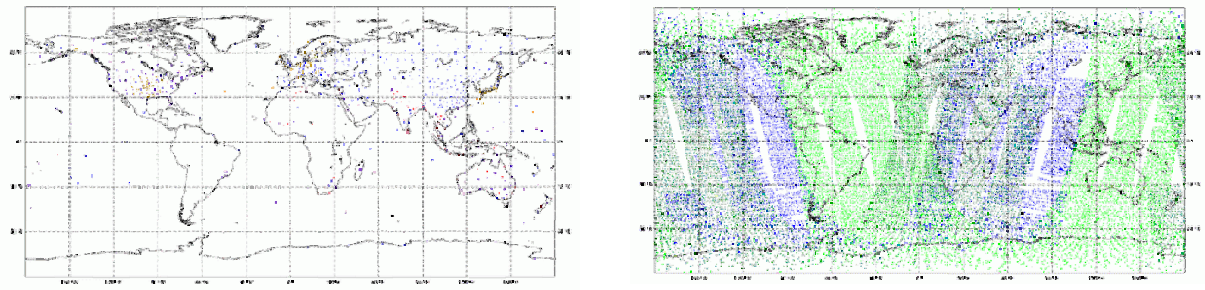


Fig. 1a: Two initially similar free-running forecasts (trajectories A and B) showing sensitive dependence on initial conditions ('chaos'). After a point in time the trajectories diverge. After this point, it might be found that neither is close to the true trajectory. Feeding-in observations (dots) using DA (trajectory C) can help keep the model close to the 'truth'.



Radiosonde ATOVS

Fig. 1b: Example coverage of radiosonde measurements and ATOVS satellite observation locations.

B.2: Why can't we use just observations to determine the state of the atmosphere?

Early attempts to determine initial conditions of models simply interpolated obs. to grid points. This approach is severely limited:

- There are too few obs. to determine the state of the system.
- Many obs. are remotely sensed - measurements are of quantities that are indirectly related to the desired model quantities (e.g. radiances are measured by satellites in orbit). These obs. cannot be simply inserted into the model.
- Direct use of obs. doesn't take account of measurement uncertainty.
- Interpolation of obs. onto a model grid doesn't ensure consistency with the laws of physics.

Obs. are instead assimilated. DA could be viewed as an 'inverse problem'. As long as we can solve the 'forward problem' (the ability to predict the obs. from a model state) then DA solves the inverse problem of determining the model state from the obs. DA can deal with the above issues.

Section C: 3-d Var. And Operational Data Assimilation

C.1: How is data assimilation used in weather forecasting?

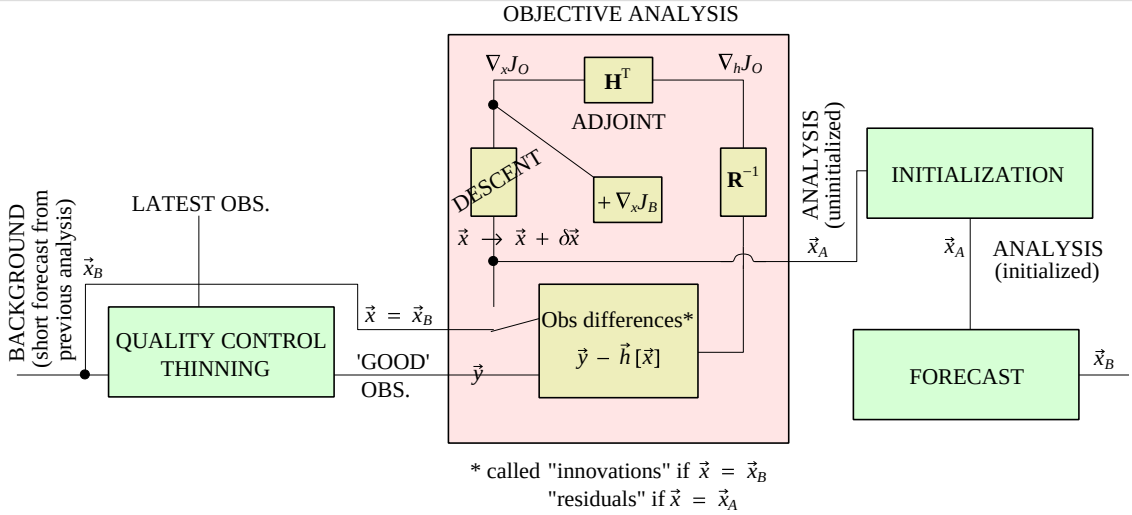


Fig. 2: The intermittent 'data assimilation cycle' showing use of a variational scheme as the data assimilation method.

C.2: What kinds of data assimilation system are there?

- Optimal interpolation (BLUE - Best Linear Unbiased Estimator).
- Kalman Filter (full KF, extended KF, ensemble KF, reduced rank KF, ...).
- Variational assimilation (1d-Var, 3d-Var., 4d-Var).

Optimal interpolation and the Kalman Filter are sometimes called 'sequential' methods; the analysis is found from an explicit (but computationally demanding) formula (see §C.7). Variational methods (Var.) obtain the analysis in an iterative fashion that minimizes a 'cost function' (see §C.3).

C.3: What is the 3d-Var. cost function?

The cost function, J, is a measure of the 'misfit' between a model state, $\vec{x}$, and other available data. The data includes (i) the observations, $\vec{y}$, and (ii) the a-priori state, $\vec{x}_B$. In Var. the aim is to find the particular $\vec{x}$ that gives minimum J (least squares). The $\vec{x}$ that achieves this minimum is called the 'analysis', $\vec{x}_A$. A simple version of the cost function is

$$J[\vec{x}] \sim (\vec{x} - \vec{x}_B)^2 + (\vec{y} - \vec{h}[\vec{x}])^2,$$

$$\sim (\vec{x} - \vec{x}_B)^{\mathrm{T}} (\vec{x} - \vec{x}_B) + (\vec{y} - \vec{h}[\vec{x}])^{\mathrm{T}} (\vec{y} - \vec{h}[\vec{x}]), \qquad (1)$$

$$J \text{ is minimized for } J[\vec{x} = \vec{x}_A].$$

- $\vec{y}$ contains the obs. values. There are p obs.
- Due to the large amount of information dealt with in DA, we use compact vector/matrix notation.
- $\vec{x}$ is called the 'state vector'. It is given in a vector space that describes the state of the forecast model (ie the physical variables, each specified on a global grid, Fig. 3). $\vec{x}$, $\vec{x}_B$ and $\vec{x}_A$ belong to the same vector space.
- The state vector has n components. The component values are often plotted as a point in n-dimensional 'state space' or 'phase space', Fig. 4.
- $\vec{x}_B$ is the 'a-priori', 'background' or 'first-guess' state. It comes from a good quality forecast.
- Part of the DA problem is to predict the obs. from a given $\vec{x}$. $\vec{h}[\vec{x}]$ is the 'observation operator' and in Var. it can be a linear or non-linear function of $\vec{x}$. The result of $\vec{h}$ exists in the same vector space as $\vec{y}$. See §E for examples.
- The observations are a source of information only about those elements of $\vec{x}$ that the function $\vec{h}$ is sensitive to. See §E for examples.
- J is exactly a (convex) parabola if $\vec{h}[\vec{x}]$ is a linear function.

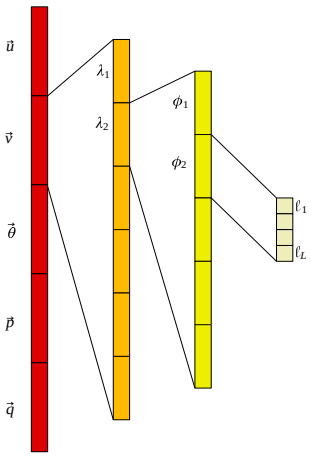


Fig. 3: The meaning of the state vector (λ, ϕ, l are longitude, latitude and vertical level). The vector has n elements in total.

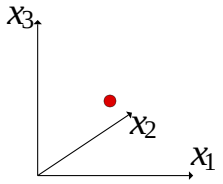


Fig. 4: State space schematic for $n = 3$.

Uncertainty of $\vec{x}_B$ and $\vec{y}$ is not dealt with in the simple form (1). Introduce 'error covariance matrices' **B** and **R** to account for uncertainty of $\vec{x}_B$ and $\vec{y}$ respectively to give the new cost function

$$J[\vec{x}] = \frac{1}{2}(\vec{x} - \vec{x}_B)^{\mathrm{T}} \mathbf{B}^{-1} (\vec{x} - \vec{x}_B) + \frac{1}{2}(\vec{y} - \vec{h}[\vec{x}])^{\mathrm{T}} \mathbf{R}^{-1} (\vec{y} - \vec{h}[\vec{x}]). \qquad (2)$$

- **R** is the *observation error covariance matrix*, Fig. 5 (and see §C.9). It is a statistical description of the random errors in $\vec{y}$. It is usually 'diagonal' (off-diagonal elements are zero) indicating that errors between each obs. are uncorrelated. Diagonal elements of **R** are the error variances of elements of $\vec{y}$.
- **B** is the *background error covariance matrix*, Fig. 6 (and see §C.9). It is a statistical description of the random errors in $\vec{x}_B$. It is an $n \times n$ matrix where matrix element i, j, $\mathbf{B}_{ij}$, describes the *error covariance* between components i and j of $\vec{x}_B$. **B** is a complicated non-sparse matrix. Diagonal elements of **B** are the error variances of elements of $\vec{x}_B$.
- Cost function (2) can be derived from Bayes' theorem (see Kalnay §5.5).
- Sometimes the J is given without the factor of 1/2.

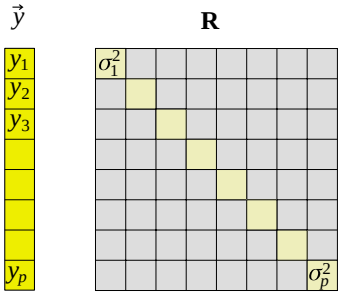


Fig. 5: The observation error covariance matrix (right) shown against the observation vector (left). Often observation errors are taken to be uncorrelated with each other and so **R** is diagonal. The diagonal matrix elements are the respective observation variances (equal to the square of the standard deviations) and the off-diagonal elements are zero. There are p observations.

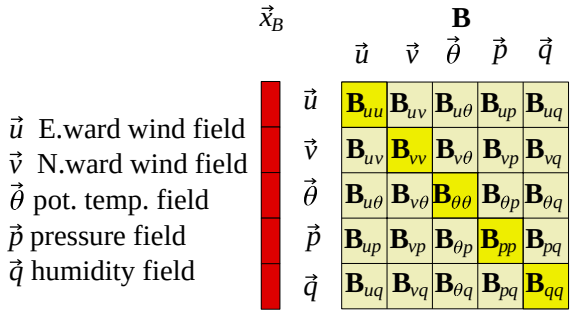


Fig. 6: The background error covariance matrix (right) for a forecast given in the state space of Fig. 3. Each square is itself a matrix here. Sub-matrices along the diagonal (deep yellow) are called 'self covariances' and off-diagonal sub-matrices are called 'multivariate covariances'.

C.4: What is '3d' about 3d-Var.?

The '3d' refers to the three spatial dimensions (e.g. longitude, latitude, height). A fourth dimension is time which is resolved in the 4d-Var. technique. Weather forecasting centres that use 3d-Var. take observations made typically within a six-hour time window. The approximation under 3d-Var is that the atmosphere does not evolve significantly within that time window (Fig. 7).

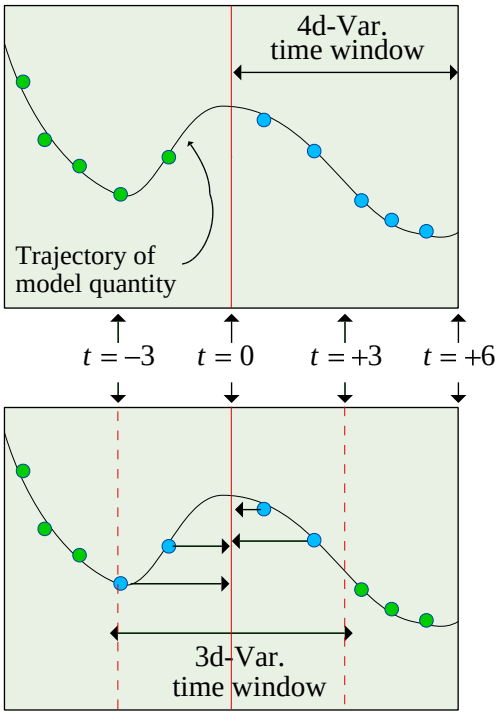
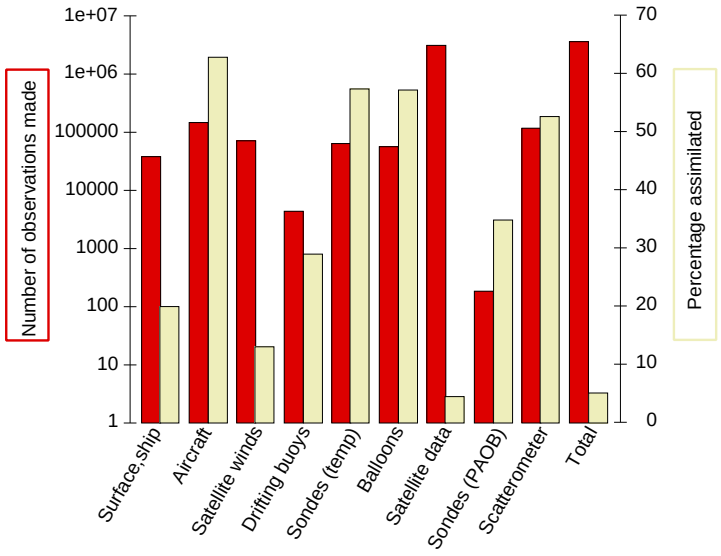


Fig. 7: Under the formulation of 4d-Var. (top), observations are used at their correct time. In 3d-Var. (bottom), the observations within a centred six-hour time period are taken as though they had been made at the same time. In each case, the analysis time is at $t = 0$.

● Observation (this cycle)
● (other cycles)

C.5: How 'large' is an operational 3d-Var. system?

- $\vec{x}$ has typically $n \sim 10^6$-10^7 elements ($\therefore$ the **B**-matrix has 10^{12}-10^{14} matrix elements!).
- $\vec{y}$ has typically $p \sim 10^5$-10^6 observations. Note that this is an order of magnitude smaller than the number of unknowns in $\vec{x}$ (hence the need to include the a-priori term ($\vec{x}_B$)).



ECMWF stats. (one cycle in June '03)
Total No. obs.: ~ 70,000,000
Total No. assimilated: ~ 3,500,000
(only 5%!)

Fig. 8: Typical numbers of observations made by instruments (red) and assimilated in the ECMWF global data assimilation system (yellow).

1993-1999 VAR coding took 42 person-years from 35 different people.

March 2001

	Subroutines, modules etc.	*Lines*
3D-Var	976	338973
PF & adjoint models (converting 3D-Var to 4D-Var)	156	87412
Obs processing & general utilities	1085	277690
Unified Model (vn5.1)	2037	522624

Fig. 9: The amount of computer code written for the Met Office Var. system is comparable to that of the Met Office forecast model. (A. Lorenc, Oxford RAL Spring School Lecture, 2001.)

C.6: How many iterations are required to minimize J?

The cost function is minimized iteratively using a descent algorithm (see §F). The starting point is $\vec{x} = \vec{x}_B$. Let J_B and J_O be the background and observation terms respectively in (2).

- As the iterations advance, J reduces, J_O reduces, but J_B increases (Fig. 10).
- The value of J at the minimum is necessarily positive (and non-zero). If the error covariance matrices are accurate descriptions of the true error statistics, if there are no biases in $\vec{x}_B$ and $\vec{y}$, if the forward operators are accurate, and if the variational assimilation has converged, then we should expect the Bennet-Talagrand result to hold that

$$J[\vec{x} = \vec{x}_A] \sim \frac{p}{2}. \qquad (3)$$

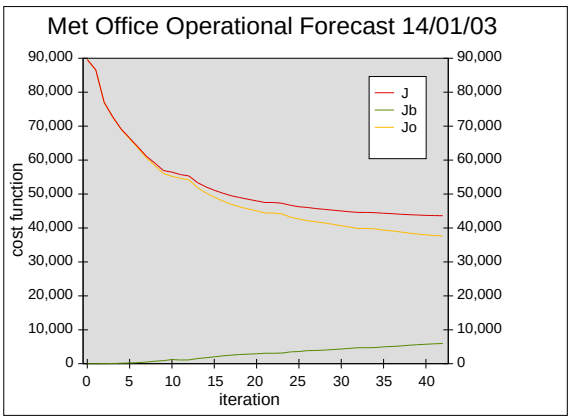


Fig 10: Value of the cost function and its components as a function of iteration for Met Office 3d-Var.

C.7: How is Var. related to the optimal interpolation formula?

A method of deriving the OI formula originates from the cost function. Even though $\vec{h}$ in (2) can be non-linear, here we will first approximate it by linearization about the $\vec{x}_B$.

$$\text{Let} \quad \vec{x} = \vec{x}_B + \delta\vec{x}, \tag{4}$$

$$\text{then} \quad \vec{h}[\vec{x}_B + \delta\vec{x}] \approx \vec{h}[\vec{x}_B] + \mathbf{H}\delta\vec{x}. \tag{5}$$

$\mathbf{H}$ is a matrix which represents the linearization of $\vec{h}$ about $\vec{x}_B$. (5) is a Taylor expansion of $\vec{h}$ about $\vec{x}_B$ to first order where $\mathbf{H}$ is the first derivative (called the 'Jacobian'),

$$\mathbf{H} = \left.\frac{\partial\vec{h}}{\partial\vec{x}}\right|_{\vec{x}_B}, \tag{6}$$

$$\text{which is a matrix notation for the elements} \quad \mathbf{H}_{ij} = \frac{\partial h_i}{\partial x_j} \quad (1 \leqslant i \leqslant p, \quad 1 \leqslant j \leqslant n). \tag{7}$$

Substitute (4)-(5) into (2), and rearrange

$$J = \frac{1}{2}\delta\vec{x}^{\mathrm{T}}\mathbf{B}^{-1}\delta\vec{x} + \frac{1}{2}(\vec{y} - \vec{h}[\vec{x_B}] - \mathbf{H}\delta\vec{x})^{\mathrm{T}}\mathbf{R}^{-1}(\vec{y} - \vec{h}[\vec{x_B}] - \mathbf{H}\delta\vec{x}),$$

$$= \frac{1}{2}\delta\vec{x}^{\mathrm{T}}\mathbf{B}^{-1}\delta\vec{x} + \frac{1}{2}(\mathbf{H}\delta\vec{x} - \{\vec{y} - \vec{h}[\vec{x_B}]\})^{\mathrm{T}}\mathbf{R}^{-1}(\mathbf{H}\delta\vec{x} - \{\vec{y} - \vec{h}[\vec{x_B}]\}).$$

J is minimized at the analysis, $\vec{x}_A$, where $\nabla_x J = 0$

$$\nabla_x J[\delta\vec{x}=\delta\vec{x}_A] = \mathbf{B}^{-1}\delta\vec{x}_A + \mathbf{H}^{\mathrm{T}}\mathbf{R}^{-1}(\mathbf{H}\delta\vec{x}_A - \{\vec{y} - \vec{h}[\vec{x_B}]\}) = 0,$$

(see §D.1 and §D.2 to derive this gradient expression), where $\vec{x}_A = \vec{x}_B + \delta\vec{x}_A$. This expression can be rearranged for $\delta\vec{x}_A$

$$(\mathbf{B}^{-1} + \mathbf{H}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{H})\,\delta\vec{x}_A = \mathbf{H}^{\mathrm{T}}\mathbf{R}^{-1}(\vec{y} - \vec{h}[\vec{x_B}]),$$

$$\delta\vec{x}_A = \vec{x}_A - \vec{x}_B = (\mathbf{B}^{-1} + \mathbf{H}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{H})^{-1}\mathbf{H}^{\mathrm{T}}\mathbf{R}^{-1}(\vec{y} - \vec{h}[\vec{x_B}]). \tag{8}$$

This equation can be written in a different way by using the following Sherman-Morrison-Woodbury formula (see the problem sheet, Q5)

$$(\mathbf{B}^{-1} + \mathbf{H}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{H})\,\mathbf{B}\mathbf{H}^{\mathrm{T}} = \mathbf{H}^{\mathrm{T}}\mathbf{R}^{-1}(\mathbf{R} + \mathbf{H}\mathbf{B}\mathbf{H}^{\mathrm{T}}), \tag{9}$$

which can be proven easily. It is straightforward to rearrange (9) to resemble the string of matrix operators that are present in (8)

$$(\mathbf{B}^{-1} + \mathbf{H}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{H})^{-1}\mathbf{H}^{\mathrm{T}}\mathbf{R}^{-1} = \mathbf{B}\mathbf{H}^{\mathrm{T}}(\mathbf{R} + \mathbf{H}\mathbf{B}\mathbf{H}^{\mathrm{T}})^{-1}, \tag{10}$$

making (8) into an equivalent form

$$\vec{x}_A - \vec{x}_B = \mathbf{B}\mathbf{H}^{\mathrm{T}}(\mathbf{R} + \mathbf{H}\mathbf{B}\mathbf{H}^{\mathrm{T}})^{-1}(\vec{y} - \vec{h}[\vec{x_B}]). \tag{11}$$

(11) is the Optimal Interpolation (OI) or Best Linear Unbiased Estimator (BLUE) formula, derived using the 'max. likelihood' (or 'min. cost') method. Since OI and Var. are equivalent when the forward model is linear (ie when (5) holds exactly), (11) can be used to understand how Var. works.

C.8: Why is 3d-Var. favoured over optimal interpolation?

Even though OI and Var. are equivalent when the forward model is linear, there are differences in the practical implementation of the methods. 3d-Var. has advantages over OI:

- 3d-Var. is more efficient than OI. For OI to be applicable to operational weather forecasting, it finds analyses for separate patches of the globe, which are then sewn together. Var., on the other hand, allows a truly global solution.

- With techniques shown in §G, 3d-Var. does not, unlike OI, have to explicitly invert large matrices.
- 3d-Var. can handle **B** in a more accurate way than OI.
- 3d-Var. can deal with indirect observations more easily than OI (indirect observations often have non-linear forward models, $\vec{h}[\vec{x}]$).
- 3d-Var. is a stepping stone to 4d-Var. 4d-Var is very similar to 3d-Var. (it shares many benefits), but has an additional forecast step as part of the forward model.
- The benefits of 3d-Var. have been demonstrated (Fig. 11).

TABLE 1. % REDUCTION IN RMS FIT OF OBSERVATIONS TO ANALYSIS (T+0) AND BACKGROUND (T+6) IN 3DVAR COMPARED WITH AC SCHEME IN THE NORTHERN HEMISPHERE IN JULY 98 TRIAL

	Temperature		Height or PMSL		Vector Wind		Relative Humidity	
Level	T+0	T+6	T+0	T+6	T+0	T+6	T+0	T+6
100hPa	-5.5	-3.3	-0.1	-3.2	15.2	5.3		
250hPa	0.8	0.0	4.8	2.5	16.8	4.9		
500hPa	5.5	2.7	3.5	5.4	14.4	3.7	5.6	2.9
700hPa	7.2	3.4	2.1	5.2	15.4	2.8	5.7	2.5
850hPa	6.6	1.4	1.4	3.7	9.1	1.8	2.9	1.5
Surface	-1.5	-0.7	6.8	-0.2	1.2	0.6		

Radiosondes TEMP reports used for Upper levels and land SYNOP reports used for Surface

Fig. 11: Performance of the Met Office 3d-Var. scheme for operational weather forecasting vs. the old Analysis Correction (AC) scheme. The AC scheme is a flavour of OI. Taken from Lorenc et al., 2000.

C.9: Why do we need to worry about the error covariance matrices?

Errors are a fundamental consideration in DA: *all models are wrong and all observations are inaccurate*. The assimilation should therefore take into account that $\vec{x}_B$ and $\vec{y}$ are known imperfectly.

There are many types of error. We assume that errors are random in nature, and follow statistically a normal (Gaussian) distribution. This is the assumption behind the cost function (2) (see Kalnay §5.5 to derive the cost function from probability distributions via Bayes' theorem).

- An error covariance matrix is a many-variable generalization of a variance.
- The cost function penalizes according to the 'distance' (in phase space) between the input data (input data being $\vec{x}_B$ or $\vec{y}$) and the assimilation's version of that data ($\vec{x}$ or $\vec{h}[\vec{x}]$). The error covariance matrices define a non-Euclidean norm.
- We can understand this more easily if we consider the error covariance matrices to be diagonal (ie that errors between components of $\vec{x}_B$ and between components of $\vec{y}$ are uncorrelated).
- If **R** is diagonal (as it usually is), the observation term in (2) becomes

$$J_O = \frac{1}{2}(\vec{y} - \vec{h}[\vec{x}])^{\mathrm{T}} \mathbf{R}^{-1} (\vec{y} - \vec{h}[\vec{x}]) = \frac{1}{2} \sum_{i=1}^{p} \frac{(y_i - h_i[\vec{x}])^2}{\mathbf{R}_{ii}}.$$

- The above is a sum of squares, weighted by the inverse variances $\mathbf{R}_{ii}^{-1}$. If a particular observation, y_i is known very well (ie small $\mathbf{R}_{ii}$, large $\mathbf{R}_{ii}^{-1}$) then any deviations from the model's predicted value, $y_i - h_i[\vec{x}]$, will suffer a very large penalty, and so $h_i[\vec{x}]$ will be strongly constrained to y_i. In the opposite regime, if y_i is known only very poorly (ie large $\mathbf{R}_{ii}$, small $\mathbf{R}_{ii}^{-1}$), then deviations will suffer only a very small penalty, and so $h_i[\vec{x}]$ will be only weakly constrained to y_i.
- The errors in **R** originate from a combination of known instrument errors and representivity errors in the ability of $\vec{h}$ to predict the observational values.
- Similar arguments apply to the background term. If **B** is diagonal (for illustrative purposes), the background term in (2) becomes

$$J_B = \frac{1}{2}(\vec{x} - \vec{x}_B)^T \mathbf{B}^{-1}(\vec{x} - \vec{x}_B) = \frac{1}{2}\sum_{i=1}^{n}\frac{(x_i - x_{Bi})^2}{B_{ii}}.$$

- The variational assimilation will be constrained more strongly to fit those elements of the background state that are known with smaller error than those with a larger error.
- It is a poor approximation to assume that **B** is a diagonal matrix. Off-diagonal elements have an information-spreading effect. In the OI equation (11), **B** is the last operator that acts (just like a convolution) to give the analysis increment. This is seen in 'single-observation' experiments in Var. (Fig. 12).
- The analysis also has uncertainties, described by an error covariance matrix, $\mathbf{P}_A$. There are many equivalent forms of $\mathbf{P}_A$. One form (12) is the inverse of the Hessian matrix. The Hessian is the second derivative of the cost function (see §D.3 and §D.4).

$$\mathbf{P}_A = (\mathbf{B}^{-1} + \mathbf{H}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{H})^{-1}. \qquad (12)$$

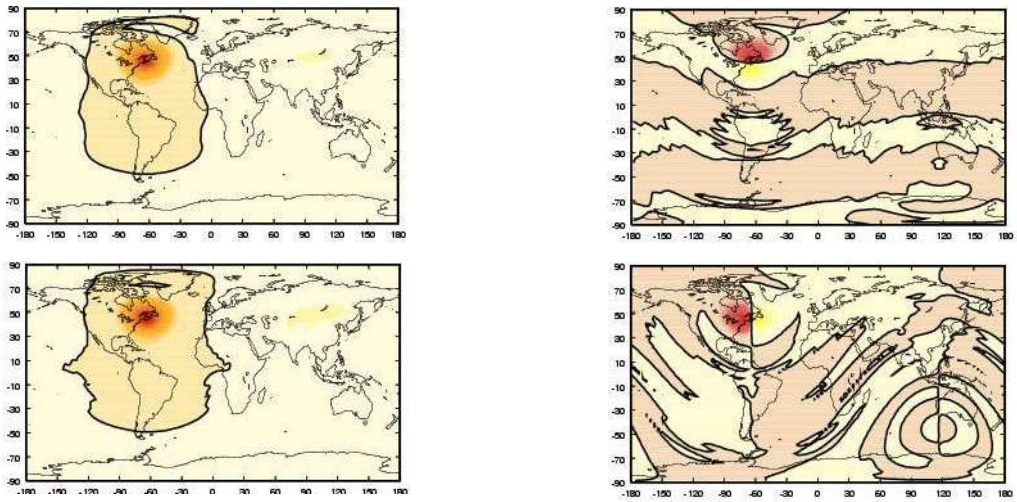


Fig. 12: Analysis increments in Var., $\delta\vec{x}_A$, due to assimilation of a pressure obs. made over the E coast of N America. Pressure (top left), potential temperature (bottom left), zonal wind (top right), meridional wind (bottom right). (11) says that these results are proportional to a column of **B**.

C.10: How can the analysis error covariance be derived?

This result, and alternative forms for $\mathbf{P}_A$ can be derived. First define errors in each quantity as a deviation from the 'truth', $\vec{x}_t$, and write outer product expressions for their error covariance matrices

$$\begin{aligned} \vec{x}_A &= \vec{x}_t + \vec{\varepsilon}_A, & \mathbf{P}_A &= \langle \vec{\varepsilon}_A \vec{\varepsilon}_A^T \rangle, \\ \vec{x}_B &= \vec{x}_t + \vec{\varepsilon}_B, & \mathbf{B} &= \langle \vec{\varepsilon}_B \vec{\varepsilon}_B^T \rangle, \\ \vec{y} &= \vec{h}[\vec{x}_t] + \vec{\varepsilon}_y, & \mathbf{R} &= \langle \vec{\varepsilon}_y \vec{\varepsilon}_y^T \rangle. \end{aligned}$$

The optimal interpolation formula can be applied to find that

$$\mathbf{P}_A = (\mathbf{I} - \mathbf{KH})\mathbf{B}, \qquad (13)$$

where $\mathbf{K} = \mathbf{BH}^{\mathrm{T}}(\mathbf{R} + \mathbf{HBH}^{\mathrm{T}})^{-1}$ (see the problem sheet, Q2). The analysis error covariance (13) is found to be the background error reduced by **KHB** due to the introduction of observational information (combining obs. with the background state reduces uncertainty). Equation (13) may be manipulated further to give (12), which is the inverse Hessian (see the problem sheet, Q2).

Section D. The Gradient And Hessian Of The Cost Function

D.1: What is the gradient vector?

The gradient vector has already been used in the derivation of the OI formula in §C.7. This vector is the derivative of J in (2) with respect to each element of $\vec{x}$.

$$\nabla_x J = \frac{\partial J}{\partial \vec{x}} = \begin{pmatrix} \partial J / \partial x_1 \\ \partial J / \partial x_2 \\ \dots \\ \partial J / \partial x_n \end{pmatrix}. \tag{14}$$

There are n elements of $\vec{x}$, and so $\nabla_x J$ also has n elements. Sometimes gradient vectors are called 'adjoint vectors' or 'sensitivities'.

D.2 How can the gradient vector be calculated?

The simplest means of computing the gradient vector is to use finite differences

$$\nabla_x J \approx \begin{pmatrix} (J[x_1 + \delta_1] - J[x_1 - \delta_1])/2\delta_1 \\ (J[x_2 + \delta_2] - J[x_2 - \delta_2])/2\delta_2 \\ \dots \\ (J[x_n + \delta_n] - J[x_n - \delta_n])/2\delta_n \end{pmatrix}.$$

This centred difference requires $2n$ evaluations of J and is inefficient. It is better to find the gradient analytically. This is found to be

$$\nabla_x J = \mathbf{B}^{-1}(\vec{x} - \vec{x}_B) - \mathbf{H}^{\mathrm{T}}\mathbf{R}^{-1}(\vec{y} - \vec{h}[\vec{x}]), \tag{15}$$

which is derived in the problem sheet, Q1. The 'transpose' of $\mathbf{H}$, denoted $\mathbf{H}^{\mathrm{T}}$, is sometimes called the 'adjoint' of $\mathbf{H}$ (while $\mathbf{H}$ on its own is sometimes called the 'forward' operator). Adjoint operators are important in Var. Note: $\mathbf{H}^{\mathrm{T}}$ is not the inverse of $\mathbf{H}$.

- The gradient is a key quantity used in the descent algorithm that minimizes J (see §F).
- It needs to be evaluated many times during the Var. algorithm (ref. Fig. 2).

D.3: What is the Hessian matrix?

The Hessian, $\mathbf{A}$, is the symmetric $n \times n$ matrix of second derivative of J calculated with respect to $\vec{x}$.

$$\mathbf{A} = \frac{\partial^2 J}{\partial \vec{x}^2} = \begin{pmatrix} \partial^2 J / \partial x_1^2 & \partial^2 J / \partial x_1 \partial x_2 & \dots & \partial^2 J / \partial x_1 \partial x_n \\ \partial^2 J / \partial x_2 \partial x_1 & \partial^2 J / \partial x_2^2 & \dots & \partial^2 J / \partial x_2 \partial x_n \\ \dots & \dots & \dots & \dots \\ \partial^2 J / \partial x_n \partial x_1 & \partial^2 J / \partial x_n \partial x_2 & \dots & \partial^2 J / \partial x_n^2 \end{pmatrix}. \tag{16}$$

- The Hessian must be positive-definate and non-singular for a unique minimum of J to exist (see the problem sheet, Q3).
- The Hessian matrix is too large to be computed for an operational Var. systems (see §C.5), but it is a useful object to understand.
- Its inverse is the error covariance of the analysis (see §C.10).
- The Hessian is used in the Newton algorithm to minimize J (see §F.3).

D.4: How can the Hessian be derived?

The Hessian can be found to have the form

$$\mathbf{A} = \mathbf{B}^{-1} + \mathbf{H}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{H}, \qquad (17)$$

which is derived in the problem sheet, Q1. The inverse of the Hessian appeared in §C.10 (the error covariance matrix of the analysis, $\mathbf{P}_A$ in (12)).

Section E: Example Observation Operators

E.1: Interpolation of temperature in a single column

Let a single column model consist of four levels. Each level has a height, z_i^m and carries temperature, T_i^m (Fig. 13). Two temperature measurements are made by a radiosonde on a weather balloon at positions between the levels. What is the forward operator, $\vec{h}$, the Jacobian, $\mathbf{H}$, and its adjoint $\mathbf{H}^T$?

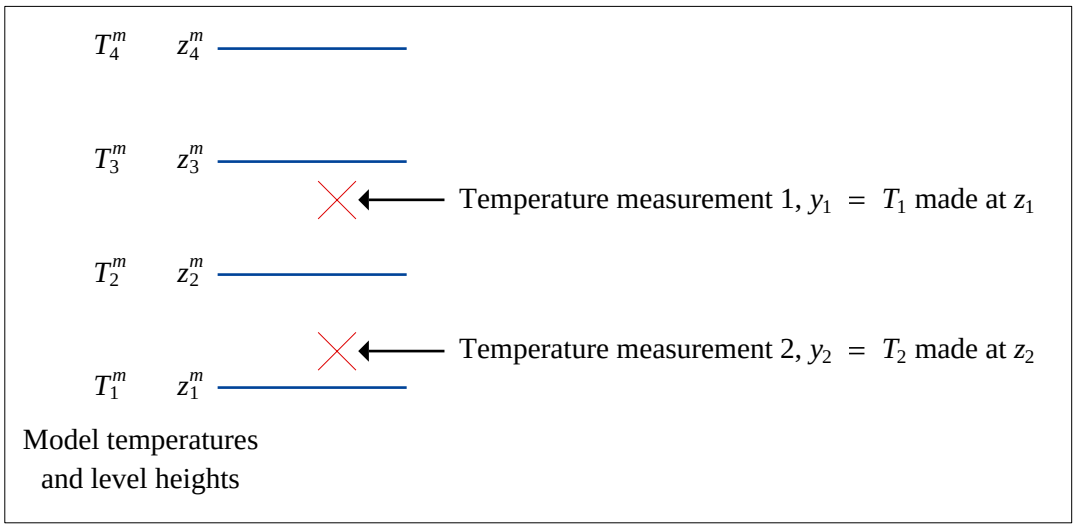


Fig. 13: The model levels and the observations.

The model state vector ($\vec{x}$) and the observation vector ($\vec{y}$) are,

$$\vec{x} = \begin{pmatrix} T_1^m \\ T_2^m \\ T_3^m \\ T_4^m \end{pmatrix}, \qquad \vec{y} = \begin{pmatrix} T_1 \\ T_2 \end{pmatrix}.$$

The forward model is found by e.g. linear interpolation,

$$\vec{h}[\vec{x}] = \begin{pmatrix} T_2^m + \frac{T_3^m - T_2^m}{z_3^m - z_2^m}(z_1 - z_2^m) \\ T_1^m + \frac{T_2^m - T_1^m}{z_2^m - z_1^m}(z_2 - z_1^m) \end{pmatrix} = \begin{pmatrix} \alpha T_2^m + \beta T_3^m \\ \gamma T_1^m + \eta T_2^m \end{pmatrix}.$$

This operator is linear. The Jacobian and its adjoint are found by evaluating (7).

$$\mathbf{H} = \begin{pmatrix} 0 & \alpha & \beta & 0 \\ \gamma & \eta & 0 & 0 \end{pmatrix}, \qquad \mathbf{H}^{\mathrm{T}} = \begin{pmatrix} 0 & \gamma \\ \alpha & \eta \\ \beta & 0 \\ 0 & 0 \end{pmatrix}.$$

- If these observations (only) were assimilated, then there would be no observational information about T_4^m, as it plays no part in the observation operator.
- This kind of observation operator is used in part of operational 3d-Var. systems.

E.2: Non-linear forward operator (radiative emission)

All bodies at a temperature above absolute zero emit thermal radiation. In this example, the radiation from a layer of the atmosphere is monitored by satellite. A forecast model represents this layer of the atmosphere with grid boxes and carries temperature in each (Fig. 14). A flux measurement is made above box 2. What is the forward operator, $\vec{h}$, the Jacobian, **H**, and its adjoint $\mathbf{H}^T$? Radiation flux, F, is related to layer temperature, T, by the Stefan-Boltzmann Law,

$$F = \kappa T^4,$$

where κ is the Stefan-Boltzmann constant.

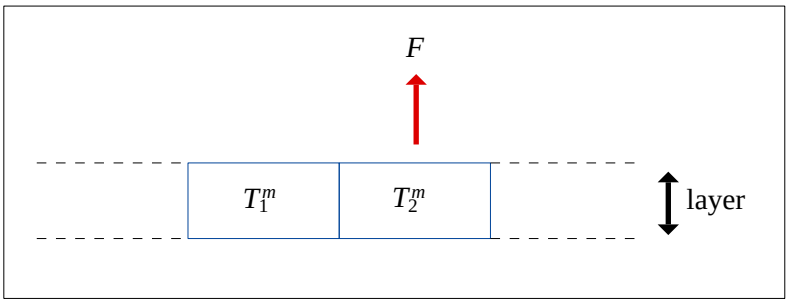


Fig. 14: Two grid boxes making up a layer of the atmosphere whose thermal radiation is being monitored by a satellite instrument.

The model state vector ($\vec{x}$) and the observation vector ($\vec{y}$) are,

$$\vec{x} = \begin{pmatrix} T_1^m \\ T_2^m \end{pmatrix}, \qquad \vec{y} = \begin{pmatrix} F \end{pmatrix}.$$

The forward model is,

$$\vec{h}[\vec{x}] = \begin{pmatrix} \kappa (T_2^m)^4 \end{pmatrix}.$$

The Jacobian is found by evaluating (7), and its adjoint follows,

$$\mathbf{H} = \frac{\partial \vec{h}}{\partial \vec{x}} = \begin{pmatrix} \partial h_1 / \partial x_1 & \partial h_1 / \partial x_2 \end{pmatrix} = \begin{pmatrix} 0 & 4\kappa (T_2^m)^3 \end{pmatrix}, \qquad \mathbf{H}^{\mathrm{T}} = \begin{pmatrix} 0 \\ 4\kappa (T_2^m)^3 \end{pmatrix}.$$

- There is no observational information about T_1^m here, as it plays no part in the obs. operator.
- Operators that predict the thermal emission of radiation to space from a column of the atmosphere are used in 3d-Var. The obs. operators deal with similar physics expressed via radiative transfer equations.
- One group of satellites providing data for operational assimilation are the ATOVS satellites (ATOVS: Advanced TIROS Operational Vertical Sounder, TIROS: Television Infrared Observational Satellite).

Section F: Minimization (or Descent) Algorithms

F.1: What is a minimization (or descent) algorithm and what is the geometric interpretation of the gradient vector?

- A minimization algorithm finds the argument ($\vec{x}$) of a scalar function (J) that gives its smallest value. Var. systems use such an algorithm.
- Algorithms are usually of an iterative (step-by-step) nature. In Var., $\vec{x}_B$ is the starting point.
- Three algorithms are shown here (there are other variants).
- Many algorithms are based on J being quadratically related to $\vec{x}$ (or approximately so). J in (2) is exactly quadratic if $\vec{h}[\vec{x}]$ is a linear function - we assume that any non-linear operators are only marginally non-linear and so the quadratic algorithms are applicable.

- The gradient vector is a key input into descent algorithms. It is required at each iteration.
- The algorithms work in *n*-dimensional state space (where *n* can be large). Here, two dimensional state space is shown schematically.
- The gradient vector, evaluated at a point in state space, points in the direction of steepest ascent. The negative gradient points in the direction of steepest descent (Fig. 15).

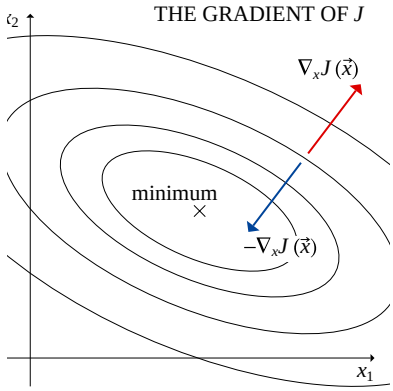
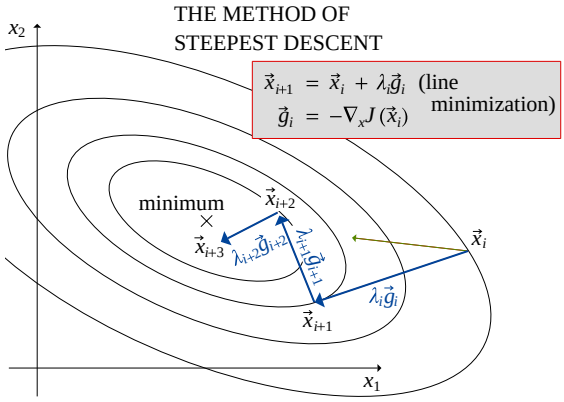


Fig. 15: Geometric representation of the gradient vector (red), and its negative (blue) in state space. The curves are contours of constant *J*.

F.2: What is the method of steepest descent?

- The method of steepest descent is the simplest and most intuitive algorithm.
- Move in the direction of steepest descent until *J* in minimized along that line (Fig. 16) (this 'line minimization' gives the 'λ_i' in Fig. 16). Repeat until sufficiently close to the minimum.
- This algorithm is inefficient, especially when the aspect ratio of the contour shape is large.
- The aspect ratio is related to the conditioning of the Hessian matrix (§G).

Fig. 16: Schematic of the method of steepest descent. The blue arrows show the path of the algorithm from one iteration to the next. The green arrow is the direction from the starting place to the minimum (in practice it is unknown).



F.3: What is the Newton algorithm (NA)?

- The min. can be reached in one iteration using the NA if *J* is exactly quadratic.
- The Hessian matrix needs to be inverted, and so the NA is impracticable for large *n*.
- The NA can be derived from the many variable Taylor expansion of *J* to second order.

Assuming *J* is exactly quadratic (ie that $\vec{h}[\vec{x}]$ is exactly linear), the Taylor expansion giving $J[\vec{x}_{i+1}]$ can be written as follows given three pieces of information evaluated at $\vec{x}_i$: (i) $J[\vec{x}_i]$, (ii) $\nabla_x J[\vec{x}_i]$ and (iii) the Hessian, **A**,

$$J[\vec{x}_{i+1}] = J[\vec{x}_i] + (\nabla_x J[\vec{x}_i])^{\mathrm{T}} (\vec{x}_{i+1} - \vec{x}_i) + \frac{1}{2} (\vec{x}_{i+1} - \vec{x}_i)^{\mathrm{T}} \mathbf{A} (\vec{x}_{i+1} - \vec{x}_i). \quad (18)$$

Differentiate with respect to $\vec{x}_{i+1}$,

$$\nabla_x J[\vec{x}_{i+1}] = \nabla_x J[\vec{x}_i] + \mathbf{A} (\vec{x}_{i+1} - \vec{x}_i), \quad (19)$$

((19) can be derived by expanding the matrix notation of (18), differentiating with respect to an individual component of $\vec{x}_{i+1}$, and then restoring the matrix notation - similar to the analysis of §D.2). Set (19) to zero (for the turning point at the minimum) and rearrange to give $\vec{x}_{i+1}$,

$$\vec{x}_{i+1} = \vec{x}_i - \mathbf{A}^{-1}\nabla_x J\,[\vec{x}_i]\,. \qquad (20)$$

$\vec{x}_{i+1}$ minimises J (and so is the analysis state). This result is identical to the OI formula (11), if the Hessian (17) and gradient (22) are substituted into (20).

F.4: What is the conjugate gradient algorithm (CGA)?

- The CGA is similar in principle to the method of steepest descent, but is more efficient.
- Unlike the Newton algorithm, it does not require the Hessian.
- Instead of using the negative gradient, $\vec{g}$, as the 'search direction', the CGA uses a modified search direction, $\vec{h}$ (Fig. 17). (N.B. $\vec{h}$ here is not the forward model.)
- The CGA is widely used.
- The algorithm's equations are not derived here.

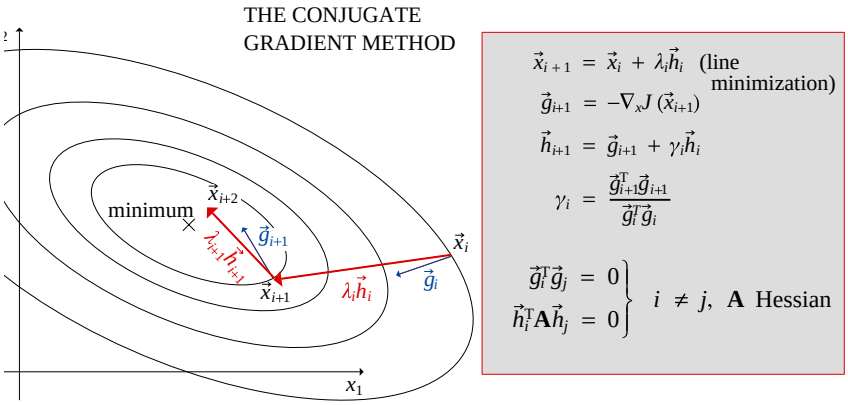


Fig. 17: Schematic of the conjugate gradient algorithm. The blue arrows are the steepest descent directions and the red arrows show the actual path of the algorithm from one iteration to the next.

Section G: Preconditioning And Control Variable Transforms

G.1: What is undesirable about minimizing (2) directly?

- The gradient $\nabla_x J$ is needed in Var. in the descent algorithm, and $\mathbf{B}^{-1}$ is needed to evaluate the gradient (22).
- **B** is far too large to store, let alone invert.
- In Var., **B** and $\mathbf{B}^{-1}$ can be approximated using 'control variable transforms'.
- Control variable transforms achieve many things:
  - Avoids the need to deal with the very large **B**-matrix.
  - Introduces a 'model' of **B** that contains the statistics and physics of forecast errors (e.g. geostrophic balance properties as in Fig. 12).
  - Preconditions the minimization problem to make it converge more quickly.
  - Allows Var. to work effectively.

G.2 What is the control variable transform that preconditions the problem?

- Make a change of variable, $\vec{x} \rightarrow \vec{\chi}$ such that the background errors in the $\vec{\chi}$-representation have a very simple structure and are well conditioned (see §G.3).

- Substitute the following into (2)

$$\text{Let} \quad \vec{x} - \vec{x}_B = \mathbf{U}\vec{\chi}, \tag{21}$$

$$\text{giving} \quad J[\vec{\chi}] = \frac{1}{2}\vec{\chi}^T\mathbf{U}^T\mathbf{B}^{-1}\mathbf{U}\vec{\chi} + \frac{1}{2}(\vec{y} - \vec{h}[\mathbf{U}\vec{\chi} + \vec{x}_B])^T\mathbf{R}^{-1}(\vec{y} - \vec{h}[\mathbf{U}\vec{\chi} + \vec{x}_B]). \tag{22}$$

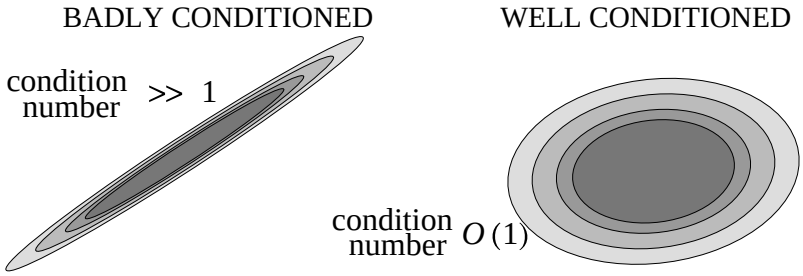
- Here, $\mathbf{U}^T\mathbf{B}^{-1}\mathbf{U}$ in the first term is the inverse of the background error covariance matrix expressed in the new $\vec{\chi}$-representation.
- Design **U** such that $\mathbf{U}^T\mathbf{B}^{-1}\mathbf{U} = \mathbf{I}$. If the **U** that achieves this can be found, the implied **B** can be found by rearranging this expression, $\mathbf{B} = \mathbf{U}\mathbf{U}^T$. The background term in the cost function then 'loses' the complicated error covariance matrix as (22) becomes,

$$J[\vec{\chi}] = \frac{1}{2}\vec{\chi}^T\vec{\chi} + \frac{1}{2}(\vec{y} - \vec{h}[\mathbf{U}\vec{\chi} + \vec{x}_B])^T\mathbf{R}^{-1}(\vec{y} - \vec{h}[\mathbf{U}\vec{\chi} + \vec{x}_B]). \tag{23}$$

- In (2) minimization is done by varying $\vec{x}$ ($\vec{x}$ was the 'control variable'). In the preconditioned problem (23), minimization is done by varying $\vec{\chi}$ ($\vec{\chi}$ is the new control variable) and the **U**-operator is the 'control variable transform'.
- In (23) **B** has not disappeared - it has been absorbed into **U**.
- In the preconditioned problem, the analysis will be $\vec{\chi}_A$ that minimizes (23). In the $\vec{x}$-representation, the analysis is found from (21), $\vec{x}_A = \mathbf{U}\vec{\chi}_A + \vec{x}_B$.
- The problem in terms of $\vec{\chi}$ (23) is better conditioned than that in terms of $\vec{x}$ (2).
- The design of the **U**-transform is a whole new lecture series!

G.3: What is meant by 'better conditioned'?

- Geometrically, the condition number is the aspect ratio of the *J* contours (Fig. 18). The shape of the contours (and hence the aspect ratio) is determined from the Hessian.



•**Fig. 18**: Contours of *J* illustrating a high conditioning number (left) and a low conditioning number (right).

- The Hessian in terms of $\vec{x}$ is $\mathbf{A}_x = \mathbf{B}^{-1} + \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}$ (17) and it is assumed that the contribution from $\mathbf{B}^{-1}$ dominates its properties (such as the condition number).
- The Hessian in terms of $\vec{\chi}$ is $\mathbf{A}_\chi = \mathbf{I} + \mathbf{U}^T\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\mathbf{U}$ which is assumed to have a lower condition number than $\mathbf{A}_x$ and hence easier to work with.