Geoscientific
Model Development

The ABC-DA system (v1.4): a variational data assimilation system for convective-scale assimilation research with a study of the impact of a balance constraint

Ross Noel Bannister

University of Reading, Dept. of Meteorology room 3L62, Meteorology Building, Whiteknights Road,
Earley Gate, Reading RG6 6ET, UK

Correspondence: Ross Noel Bannister (r.n.bannister@reading.ac.uk)

Abstract. Following the development of the simplified atmospheric convective-scale "toy" model (the ABC model, named after its three key parameters: the pure gravity wave frequency A, the controller of the acoustic wave speed B, and the constant of proportionality between pressure and density perturbations C), this paper introduces its associated variational data assimilation system, ABC-DA. The purpose of ABC-DA is to permit quick and efficient research into data assimilation methods suitable for convective-scale systems. The system can also be used as an aid to teach and demonstrate data assimilation principles.

ABC-DA is flexible and configurable, and is efficient enough to be run on a personal computer. The system can run a number of assimilation methods (currently 3DVar and 3DF-GAT have been implemented), with user configurable observation networks. Observation operators for direct observations and wind speeds are part of the current system, and these can, for example, be expanded relatively easily to include operators for Doppler winds. A key feature of any data assimilation system is how it specifies the background error covariance matrix. ABC-DA uses a control variable transform method to allow this to be done efficiently. This version of ABC-DA mirrors many operational configurations by modelling multivariate error covariances with uncorrelated control parameters, each with special uncorrelated spatial patterns.

The software developed performs (amongst other things) model runs, calibration tasks associated with the background error covariance matrix, testing and diagnostic tasks, single data assimilation runs, and multi-cycle assimilation/forecast experiments, and it also has associated visualisation software.

As a demonstration, the system is used to tackle a scientific question concerning the role of geostrophic balance (GB) to model background error covariances between mass and wind fields. This question arises because although GB is a very useful mechanism that is successfully exploited in larger-scale assimilation systems, its use is questionable at convective scales due to the typically larger Rossby numbers where GB is not so relevant. A series of identical twin experiments is done in cycled assimilation configurations. One experiment exploits GB to represent mass–wind covariances in a mirror of an operational set-up (with use of an additional vertical regression (VR) step, as used operationally). This experiment performs badly where error accumulates over time. Two further experiments are done: one that does not use GB and another that does but without the VR step. Turning off GB impairs the performance, and turning off VR improves the performance in general. It is concluded that there is scope to further improve the way that the background error covariance matrices are represented at convective scale. Ideas for further possible developments of ABC-DA are discussed.

1 Introduction

The grid sizes of limited-area models for operational weather forecasting have become small enough to allow some convective processes to be resolved explicitly (Clark et al., 2016; Yano et al., 2018). Some leading operational models in-

clude the COSMO (COnsortium for Small-scale MOdelling) model (Baldauf et al., 2011), used at MeteoSwiss (1.1 km grid size) and at the Deutscher Wetterdienst (DWD) (2.8 km grid size); the AROME (Application of Research to Operations at Mesoscale) model (Brousseau et al., 2016), used at Météo-France (1.3 km grid size); the UKV (UK Variable resolution) model (Tang et al., 2013) (1.5 km grid size); and the WRF (US Weather Research and Forecasting) model (Schwartz and Liu, 2014) (3 km grid size). Each of these systems is invaluable in the forecasting of fine-scale weather, including that associated with convective storms, and has its own data assimilation (DA) system to estimate its initial conditions from new observations and a background state.

Apart from the capability to assimilate new high-resolution observation types, such as radar reflectivity and Doppler radial wind, the DA systems are still based on those designed for use with synoptic- and planetary-scale phenomena in mind. The convective-scale DA problem needs to account for effects that can often be safely ignored or treated approximately when dealing with large scales. These include certain dynamical properties of background state errors (namely non-hydrostatic and non-geostrophic contributions, vertical motion, multiple phases of water, strong inhomogeneity and flow dependence, and non-Gaussianity), certain properties of observation errors (namely cross-correlations), and other features associated with a small grid size (e.g. feature misalignment). There are also challenges associated with assimilating new observation types (as mentioned above, including the large volumes of data needed), the short DA time window (often 1 h or less), the compatibility of lateral boundary conditions from a coarser parent model, and questions concerning the appropriateness of allowing DA to simultaneously modify the larger-scale flows present in the convective-scale problem.

The properties of background state errors are of particular concern to this paper, although the DA system to be described can be equally applied to study other aspects of convective-scale DA, such as the exploration of strategies for high-resolution observation networks (such as those from Doppler wind instruments), or indeed some of the advanced DA methods mentioned below. In DA, the background state is traditionally assumed to be subject to random error, which is distributed according to a Gaussian distribution described by a multivariate error covariance matrix (the "**B**-matrix", e.g. Bannister, 2008a). Given that **B** is too large to store explicitly, in variational DA (Var) it is represented in the form of a "model" (Bannister, 2008b). One important means of representing **B** in a way that naturally adapts to the flow conditions is to derive a matrix implicitly from an ensemble of forecasts, which are often produced anyway for probabilistic forecasting purposes. This is the basis of the ensemble Kalman filter (e.g. Houtekamer and Zhang, 2016) and EnVar (pure ensemble-variational) formulations (e.g. Liu et al., 2008). Although information from an ensemble in principle follows the dynamical properties of the model to be reflected

in the **B** matrix used at the analysis time, the result is often corrupted by sampling error due to the small ensemble sizes (usually a few tens of members). For this reason, the **B** matrix used operationally is often still modelled according to physical insight. That insight, though, is based on traditional assumptions of (for instance) geophysical balance, whose applicability is questionable at convective scales.

Studying convective-scale DA in operational systems is burdened severely by the cost and complexity of these systems. The DA system described in this paper has been designed in the same spirit as that of the convective-scale toy model (the ABC model; Petrie et al., 2017), i.e. with an emphasis on low cost and simplicity. This DA system (together with the model code, hereafter called ABC-DA) is a multi-featured Var system suited to the ABC model. ABC-DA is actually a suite of software used not only to perform DA itself (in cycling mode if required) but also to calibrate the **B** matrix from sets of forecast data, to flexibly generate randomly perturbed data such as synthetic observations from a *truth* run (which can then be assimilated) to compute a sample of covariances implied from a chosen **B** matrix model and to perform a set of validation tests. The suite also includes sample plotting codes to help visualise and monitor the outputs, a script to build the executables, sample run scripts, and detailed user documentation. The ABC-DA numerical codes are written in Fortran 90, scripts are written in Linux Bash, and the plotting code is written in Python 2. Certain open-source software libraries are also required to compile the code.

This paper presents the scientific documentation for the system, with examples and pointers to how a user can access the software. Finally, a short study of ABC-DA is presented to investigate the impact of balance constraints in the formulation of the convective-scale **B** matrix. The paper is structured as follows. In Sect. 2 the ABC model is described, in Sect. 3 the ABC-DA system is outlined, in Sect. 4 the ABC-DA system is described in detail, in Sect. 5 a brief study of the role of geostrophic balance is presented, and in Sect. 6 the paper is summarised.

2 The ABC model

2.1 The model equations

The ABC model comprises a set of simplified partial differential equations for a two-dimensional spatial grid (x and z) plus time (t), which are based on the Euler equations. This section summarises the ABC model, and the reader is directed to Petrie et al. (2017) for the details. The model equations are as follows:

$$\frac{\partial u}{\partial t} + B\boldsymbol{u} \cdot \nabla u + C\frac{\partial \tilde{\rho}'}{\partial x} - fv = 0, \tag{1a}$$

$$\frac{\partial v}{\partial t} + B\boldsymbol{u} \cdot \nabla v + fu = 0, \tag{1b}$$

$$\frac{\partial w}{\partial t} + B\boldsymbol{u} \cdot \nabla w + C\frac{\partial \tilde{\rho}'}{\partial z} - b' = 0, \tag{1c}$$

$$\frac{\partial \tilde{\rho}'}{\partial t} + B\nabla \cdot (\tilde{\rho}\boldsymbol{u}) = 0, \tag{1d}$$

$$\frac{\partial b'}{\partial t} + B\boldsymbol{u} \cdot \nabla b' + A^2 w = 0, \tag{1e}$$

where $\boldsymbol{u} = \begin{pmatrix} u & v & w \end{pmatrix}$ is the wind vector (comprising zonal, meridional, and vertical wind components, respectively); $\tilde{\rho}$ is the scaled density variable (akin to pressure); b is the buoyancy variable (akin to temperature); f is the Coriolis parameter; g is the acceleration due to gravity; and A, B, and C are tunable parameters (see below). Primed variables indicate perturbations from a reference state defined as $b(x, z, t) = g + b_0(z) + b'(x, z, t)$ and $\tilde{\rho}(x, z, t) = 1 + \tilde{\rho}'(x, z, t)$. The model supports a range of motions, namely balanced (Rossby-like) modes and unbalanced (gravity and acoustic) modes, which have been studied in detail in Petrie et al. (2017). There are three tunable parameters: A is the pure gravity wave frequency, controlling the gravity wave speeds in the model; B modulates the advective and divergent terms in the equations, controlling the acoustic wave speeds; and C specifies the equation of state that relates pressure and density perturbations, $p' = C\rho_0\tilde{\rho}'$, where ρ_0 is a density scaling constant. In the linearised equations, only the product BC (and not B and C individually) affects the characteristics of the flow, so C also controls the acoustic wave speeds. In numerical integrations of the non-linear equations (Eq. 1), the effect of scaling C is found to be virtually indistinguishable from scaling B in terms of the patterns of forecast perturbations.

2.2 Properties of the ABC model equations

Equation (1) conserves total mass and energy, although this exact property is lost when the equations are discretised for numerical integration. The equations approximate to geostrophic and hydrostatic balance (GB and HB, respectively) when the Rossby number, $Ro = \mathcal{U}/f\mathcal{L}$, is small (where $\mathcal{U}$ is the characteristic zonal wind speed and $\mathcal{L}$ is the characteristic horizontal length scale of the motion). The GB relations are

$$-fv + C\frac{\partial \tilde{\rho}'}{\partial x} = 0, \tag{2a}$$

$$u = 0, \tag{2b}$$

and the HB relation is

$$-b' + C\frac{\partial \tilde{\rho}'}{\partial z} = 0. \tag{3}$$

These balance relations are well satisfied for motion at the large scales where Ro is small, and they are used in traditional Var schemes to model the covariances between mass, wind, and temperature perturbations in background errors. We will revisit these later in the paper.

2.3 Discretisation and integration

As reported in Petrie et al. (2017), the continuous equations (Eq. 1) have been discretised in time and space; the current implementation uses a 360×60 (horizontal × vertical) element grid with a grid box size of $1500\,\mathrm{m} \times 250\,\mathrm{m}$. Variables are stored on an Arakawa C grid in the horizontal and Charney–Phillips grid in the vertical (see Fig. 1 of Petrie et al., 2017), and periodic boundary conditions are imposed in the horizontal to avoid the need for a driving model to provide lateral boundary conditions. The integration scheme used is the split-explicit forward–backward scheme of Cullen and Davies (1991) with a main time step of $\Delta t = 4\,\mathrm{s}$.

2.4 Future developments of the model

The current version of the ABC model does not include moist processes. There is much that can be learned about convective-scale DA from a dry model, but assimilating and forecasting moisture fields is a major reason for convective-scale forecasting (Sun et al., 2014; Bannister et al., 2020). It is planned to upgrade the model to permit the advection of one or more water variables and allow condensation and evaporation processes to affect the flow. The assimilation of moisture is a complex task and so a moist ABC-DA system is expected to be very useful in that line of research.

3 Overview of the ABC-DA system

Like the ABC model, the DA system is intended to be low cost and easy to run, when compared to a operational-scale system, yet mirror many of the features and options available in operational systems. In this section we review the principles on which ABC-DA is based, which includes a definition of the mathematical notation, but we leave it to Sect. 4 to describe the details.

3.1 Variational data assimilation

Var systems construct a scalar functional (called a cost function, J) that is minimised with respect to the state $\boldsymbol{x}$ by considering observations made over a time window (indicated by an integer time index) $t \in [0, T]$:

$$J[\boldsymbol{x}] = \frac{1}{2}(\boldsymbol{x} - \boldsymbol{x}^{\mathrm{b}})\mathbf{B}^{-1}(\boldsymbol{x} - \boldsymbol{x}^{\mathrm{b}})$$

$$+ \frac{1}{2}\sum_{t=0}^{T}\left[\boldsymbol{y}(t) - \boldsymbol{y}^{\mathrm{m}}(t)\right]^{\mathrm{T}}\mathbf{R}_t^{-1}\left[\boldsymbol{y}(t) - \boldsymbol{y}^{\mathrm{m}}(t)\right]. \tag{4}$$

In Eq. (4), x represents all variables of the model state at $t = 0$ (here u, v, w, $\tilde{\rho}'$, and b') at each location in the domain, x^{b} is a special state at $t = 0$ called the background state (normally a short forecast from the previous DA), $y(t)$ is the collection of observations at time t, and $y^{m}(t)$ is the model's version of the observations computed from x. Let there be n elements in x and p_t elements in $y(t)$. Here we use the convention that quantities like x without a time argument imply the value at $t = 0$. The model's observations are found in two steps. Firstly, the state is found at time t using the model propagator: $x(t) = \mathcal{M}_{0 \to t}(x)$, which is the result of integrating Eq. (1) from times 0 to t, and then the model's observations are found using the observation operator at this time: $y^{m} = \mathcal{H}_t(x(t))$. This DA method is known as 4DVar (Dimet and Talagrand, 1986). $\mathbf{B}$ and $\mathbf{R}_t$ are covariance matrices (e.g. Kalnay, 2002) pertaining to errors in the background state and in the observations at time t, respectively. Mathematically $\mathbf{B}$ (an $n \times n$ matrix) and $\mathbf{R}_t$ (a $p_t \times p_t$ matrix) may be thought as the metrics in which deviations are measured in the cost function (i.e. measures of the precision to which the background and observations are known). The analysis x^{a} is the special state that minimises J.

The $\mathbf{B}$ and $\mathbf{R}_t$ matrices are important as they can have a profound effect on the way that observations combine with the background to yield the analysis. It is a particular challenge to use a $\mathbf{B}$ matrix that is relevant to the uncertainties in convective-scale forecasts. As $\mathbf{B}$ is usually a much larger matrix than $\mathbf{R}_t$ (operationally by orders of magnitude), it cannot practically be stored explicitly. Instead the $\mathbf{B}$ matrix is modelled, which is usually done via the technique of control variable transforms (CVTs) – see Sect. 3.4. As this modelling process is a major component of any DA system, and may require new thinking for convective-scale systems, much of the design of ABC-DA is concerned with how $\mathbf{B}$ is modelled. As a starting point for ABC-DA, the approach that is currently implemented is a conventional one (to mirror typical current operational configurations that were designed around global systems, Sect. 4.2, but still applied for convective-scale systems). This approach, though, can be adapted to accommodate new convective-scale strategies that are discussed at the end of the paper.

3.2 The incremental formulation of the problem

If $\mathcal{M}_{0 \to t}$ and $\mathcal{H}_t$ are linear functions, then Eq. (4) is a quadratic function of x and may be minimised using efficient algorithms such as a conjugate gradient-based method (Golub and Van Loan, 1996; Lewis et al., 2006). The model $\mathcal{M}_{0 \to t}$, though, is a non-linear operator, and many observations require non-linear observation operators (such as measurements of wind speed, top-of-atmosphere radiance, etc.). This leads to a non-quadratic function, which may have multiple minima. Furthermore, there are no general efficient minimising algorithms for such problems. In order to simplify the problem, the cost function is minimised by breaking it

down into a sequence of quadratic problems by iteratively linearising $\mathcal{M}_{0 \to t}$ and $\mathcal{H}_t$. This is incremental Var (Courtier et al., 1994).

Suppose that $x^{r}(t)$ is a reference trajectory satisfying $x^{r}(t) = \mathcal{M}_{t-1 \to t}(x^{r}(t-1))$, $1 \leq t \leq T$. A perturbation to this trajectory (δ prefix) at t is approximately related to a perturbation at $t - 1$ via the linear operation $\delta x(t) \approx \mathbf{M}_{t-1 \to t}\delta x(t-1)$, where the full states are $x(t) \approx x^{r}(t) + \delta x(t)$ and $x(t-1) \approx x^{r}(t-1) + \delta x(t-1)$. $\mathbf{M}_{t-1 \to t}$ is the tangent linear (or Jacobian) of $\mathcal{M}_{t-1 \to t}$ and is mathematically representable by an $n \times n$ matrix. We assume that $x^{r}(t) + \delta x(t)$ is close to $\mathcal{M}_{t-1 \to t}(x^{r}(t-1) + \delta x(t-1))$, provided that $\delta x(t-1)$ is sufficiently small. Similarly, we suppose that the observation values computed from $x^{r}(t)$ satisfy $y^{mr}(t) = \mathcal{H}_t(x^{r}(t))$. A perturbation to these reference observations is approximately related to a perturbation in $x^{r}(t)$ via the linear operation $\delta y(t) = \mathbf{H}_t \delta x(t)$. $\mathbf{H}_t$ is the tangent linear (or Jacobian) of $\mathcal{H}_t$ and is mathematically representable by a $p_t \times n$ matrix. These approximations may be summarised as the following:

$$x \approx x^{r} + \delta x, \tag{5a}$$
$$y^{m}(t) \approx y^{mr}(t) + \delta y^{m}(t), \tag{5b}$$

where

$$\delta y^{m}(t) = \mathbf{H}_t \mathbf{M}_{0 \to t} \delta x. \tag{5c}$$

When Eq. (5) and the following definitions,

$$\delta x^{b} = x^{b} - x^{r}, \tag{6a}$$
$$d(t) = y(t) - \mathcal{H}_t\left(\mathcal{M}_{0 \to t}\left(x^{r}\right)\right) = y(t) - y^{mr}(t), \tag{6b}$$

are substituted into Eq. (4), J becomes a functional of the perturbation δx instead of the full state x:

$$
\begin{aligned}
J[\delta x] = & \frac{1}{2}\left(\delta x - \delta x^{b}\right)^{\mathrm{T}} \mathbf{B}^{-1}\left(\delta x - \delta x^{b}\right) \\
& + \frac{1}{2}\sum_{t=0}^{T}[\mathbf{H}_t \mathbf{M}_{0 \to t}\delta x - d(t)]^{\mathrm{T}} \mathbf{R}_t^{-1}[\mathbf{H}_t \mathbf{M}_{0 \to t}\delta x - d(t)].
\end{aligned}
\tag{7}
$$

This is the incremental form of 4DVar. $J[\delta x]$ is exactly quadratic, allowing efficient algorithms to be used to minimise it to yield the special state δx^{a}. The iterations required to minimise Eq. (7) form an *inner loop*. The full cost function (Eq. 4) is minimised by updating the reference state, $x^{r} \to x^{r} + \delta x^{a}$, and repeating the inner loops. These iterations form the *outer loop*. In the first outer loop, x^{r} is typically set to x^{b}. This inner and outer loop procedure, though, does not necessarily find the global minimum of Eq. (4), which can lead to complications in highly non-linear systems with a long DA time window (e.g. Fabry and Sun, 2010).

Because $\mathbf{M}_{0 \to t}$ is a difficult operator to derive, the approximation that $\mathbf{M}_{0 \to t} = \mathbf{I}$ is often made. This leads to an

approximate method called 3DFGAT (3DVar First Guess at Appropriate Time; Lee et al., 2004; Lawless, 2010). For applications when it is too expensive to use $\mathcal{M}_{0 \to t}$ in the DA loops, a further approximation is made that $\boldsymbol{x}(t) = \boldsymbol{x}$ over the window. This is called 3DVar, although many systems that specify 3DVar actually use 3DFGAT.

3.3 The observations, their operators, and their error statistics

The observation operator $\mathcal{H}_t$ (and $\mathbf{H}_t$) is built to suit the range of observations assimilated. The components of $\mathcal{H}_t$ and $\mathbf{H}_t$ represent the model's version of the observations. Typical examples include simple bi-linear interpolation of grid values to an observation location, the computation of model wind speed as the root of the sum of squares of the wind components, or the evaluation of top-of-atmosphere radiance by a radiative transfer equation. The ABC-DA system currently implements observations of the first two kinds, but the system is flexible enough to support observations of any required model variable at arbitrary times and positions. The $\mathbf{R}_t$ matrices are taken to be diagonal, and have specified variances. The system could be adapted to extend any of these aspects to include more complicated observation operators, such as radiative transfer models, Doppler winds, or for correlated observation errors.

3.4 Modelling B with control variable transforms

The $\mathbf{B}$ matrix is meant to represent the covariances of errors in $\boldsymbol{x}^\mathrm{b}$. Operational-scale DA systems all share the challenge of determining and using $\mathbf{B}$ given that this $n \times n$ matrix is too large to manipulate (or even store) and is in any case unknowable. Most practical variational methods use *control variable transforms* (CVTs) to simplify this problem. Consider a vector $\delta\boldsymbol{\chi}$, which is an alternative representation of $\delta\boldsymbol{x}$ via the relation

$$\delta\boldsymbol{x} = \mathbf{U}\delta\boldsymbol{\chi}. \tag{8}$$

$\delta\boldsymbol{\chi}$ is called a *control vector*, and $\mathbf{U}$ is the CVT. The CVT is a powerful way of accounting for cross-correlations between background errors of model variables (including spatial and multivariate components). $\delta\boldsymbol{x}$ and $\delta\boldsymbol{\chi}$ have different assumed statistical properties: model space errors have covariance $\langle\delta\boldsymbol{x}\delta\boldsymbol{x}^\mathrm{T}\rangle_\mathrm{b} = \mathbf{B}$, and control variables are taken to be uncorrelated and have unit variance $\langle\delta\boldsymbol{\chi}\delta\boldsymbol{\chi}^\mathrm{T}\rangle_\mathrm{b} = \mathbf{I}$, where the b subscript indicates expectation over hypothetical background error samples. When Eq. (8) is substituted into Eq. (7), J becomes a functional of $\delta\boldsymbol{\chi}$:

$$J[\delta\boldsymbol{\chi}] = \frac{1}{2}\left(\delta\boldsymbol{\chi} - \delta\boldsymbol{\chi}^\mathrm{b}\right)^\mathrm{T}\left(\delta\boldsymbol{\chi} - \delta\boldsymbol{\chi}^\mathrm{b}\right)$$
$$+ \frac{1}{2}\sum_{t=0}^{T}[\mathbf{H}_t\mathbf{M}_{0 \to t}\mathbf{U}\delta\boldsymbol{\chi} - \boldsymbol{d}(t)]^\mathrm{T}$$
$$\mathbf{R}_t^{-1}[\mathbf{H}_t\mathbf{M}_{0 \to t}\mathbf{U}\delta\boldsymbol{\chi} - \boldsymbol{d}(t)]. \tag{9}$$

Notice that the $\mathbf{B}$ matrix has effectively disappeared from this formulation because of the assumed statistical properties of $\delta\boldsymbol{\chi}$. This is a considerable simplification, although the problem of defining the CVT remains. The $\delta\boldsymbol{\chi}$ cost function is minimised (giving the special vector $\delta\boldsymbol{\chi}^\mathrm{a}$), which then leads to $\delta\boldsymbol{x}^\mathrm{a}$ via Eq. (8). This is equivalent to minimising Eq. (7) with a background error covariance matrix equal to

$$\mathbf{B}^\mathrm{ic} = \mathbf{U}\mathbf{U}^\mathrm{T}, \tag{10}$$

which is known as the *implied background error covariance matrix*. Apart from not needing to know the $\mathbf{B}$ matrix explicitly, the minimisation problem (Eq. 9) is found to be numerically better conditioned than Eq. (7), leading to a more efficient and accurate minimisation.

It is common to work backwards here: first a CVT is proposed (based on physical principles such as those discussed in Bannister (2008b) and later in this paper), and then its ability to generate reasonable background error covariance structures is studied by looking at the implied covariances. This can be done by studying either $\mathbf{U}\mathbf{U}^\mathrm{T}$ or the analysis increments of single-observation DA experiments. Constructing $\mathbf{U}$ is one way of doing *background error covariance modelling*. We do this by defining new parameters and their spatial covariances via a proposed form like $\mathbf{U} = \mathbf{U}_\mathrm{p}\mathbf{U}_\mathrm{s}$ (Bannister, 2008b). Here $\mathbf{U}_\mathrm{p}$ is the *parameter transform* (where $\mathbf{U}_\mathrm{p}^{-1}$ transforms model variables to alternative parameters that are assumed uncorrelated using sets of balance operators as in Parrish and Derber, 1992; Gauthier et al., 1999), and $\mathbf{U}_\mathrm{s}$ is the *spatial transform* (which transforms each parameter's field to modes that are assumed to be uncorrelated, such as Fourier modes). $\mathbf{U}_\mathrm{s}$ can itself be decomposed into separate horizontal, vertical, and scaling parts; e.g. $\mathbf{U}_\mathrm{s} = \boldsymbol{\Sigma}\mathbf{U}_\mathrm{v}\mathbf{U}_\mathrm{h}$ (see Sect. 4.2.3). More complicated sequences of transforms are also possible, e.g. based on wavelets (Deckmyn and Berre, 2005). A property of the CVT approach is that $\mathbf{B}$ can be modelled even if it is singular.

3.5 The gradient of J and minimising the cost function

Equation (7) is minimised by iteratively adjusting $\delta\boldsymbol{\chi}$ until a convergence criterion is met, indicating that a point close to the minimum of $J[\delta\boldsymbol{\chi}]$ has been found. The gradient vector, $\nabla_{\delta\boldsymbol{\chi}}J$, is used with a conjugate gradient algorithm to perform this task. $\nabla_{\delta\boldsymbol{\chi}}J$ is found by differentiating $J[\delta\boldsymbol{\chi}]$:

$$\nabla_{\delta\boldsymbol{\chi}}J = \delta\boldsymbol{\chi} - \delta\boldsymbol{\chi}^\mathrm{b} + \mathbf{U}^\mathrm{T}\sum_{t=0}^{T}\mathbf{M}_{0 \to t}^\mathrm{T}\mathbf{H}_t^\mathrm{T}\mathbf{R}_t^{-1}$$
$$[\mathbf{H}_t\mathbf{M}_{0 \to t}\mathbf{U}\delta\boldsymbol{\chi} - \boldsymbol{d}(t)], \tag{11}$$

where $\boldsymbol{d}(t)$ is the difference between the observations at time t and the model's version of them based on the reference state Eq. (6b). Equation (11) requires the Jacobians $\mathbf{H}_t$ and $\mathbf{M}_{0 \to t}$, the CVT $\mathbf{U}$, and their adjoint counterparts. The evaluation of Eq. (11) can be made more efficient by the following standard algorithm.

1. Set the reference state at $t = 0$ to the background state: $x^{\mathrm{r}} = x^{\mathrm{b}}$.

2. Do the outer loop.

   (a) For the first outer loop, $\delta \chi^{\mathrm{b}} = 0$; otherwise, compute $\delta \chi^{\mathrm{b}} = \mathbf{U}^{-1} \left(x^{\mathrm{b}} - x^{\mathrm{r}} \right)$.

   (b) Compute $x^{\mathrm{r}}(t)$ over the time window, $1 \leq t \leq T$, with the non-linear model: $x^{\mathrm{r}}(t) = \mathcal{M}_{t-1 \to t} \left(x^{\mathrm{r}}(t-1) \right)$.

   (c) Compute the reference state's observations: $y^{\mathrm{mr}}(t) = \mathcal{H}_t \left(x^{\mathrm{r}}(t) \right)$.

   (d) Compute the differences: $d(t) = y(t) - y^{\mathrm{mr}}(t)$.

   (e) Set $\delta \chi = 0$ and $\delta x = 0$.

   (f) Do the inner loop.

      i. Integrate the perturbation trajectory over the time window, $1 \leq t \leq T$, with the linear forecast model: $\delta x(t) = \mathbf{M}_{t-1 \to t} \delta x(t-1)$.

      ii. Compute the perturbations to the model observations: $\delta y^{\mathrm{m}}(t) = \mathbf{H}_t \delta x(t)$.

      iii. Compute $\boldsymbol{\Delta}(t)$ vectors defined as $\boldsymbol{\Delta}(t) = \mathbf{H}_t^{\mathrm{T}} \mathbf{R}_t^{-1} \left[\delta y^{\mathrm{m}}(t) - d(t) \right]$.

      iv. Set the adjoint state $\lambda(T+1) = 0$.

      v. Integrate the following adjoint equation backwards in time, $T \geq t \geq 0$: $\lambda(t) = \boldsymbol{\Delta}(t) + \mathbf{M}_{t \to t+1}^{\mathrm{T}} \lambda(t+1)$.

      vi. Compute the gradient as follows: $\nabla_{\delta \chi} J = \delta \chi - \delta \chi^{\mathrm{b}} + \mathbf{U}^{\mathrm{T}} \lambda(0)$.

      vii. Use the conjugate gradient algorithm to adjust $\delta \chi$ to reduce the value of J.

      viii. Compute the new increment in model space using the CVT: $\delta x = \mathbf{U} \delta \chi$.

      ix. Go to step 2fi until the inner-loop convergence criterion is satisfied.

   (g) Update the reference state: $x^{\mathrm{r}} \to x^{\mathrm{r}} + \delta x$.

   (h) Go to step 2a until the outer-loop convergence criterion is satisfied. At convergence, set $x^{\mathrm{a}} = x^{\mathrm{r}}$.

3. Run a non-linear forecast from x^{a} for the background of the next cycle and longer forecasts if required.

The full procedure (adapted for 3DFGAT, where the linear model is omitted) is shown graphically in Sect. 4.7.

3.6 System tests

The system has a special test suite to check aspects of operators that are coded. Operators that have an adjoint counterpart are subject to an adjoint test to demonstrate that the adjoint has been coded correctly. This includes the linearised observation operators and components of $\mathbf{U}$. Many of these operators are subdivided into constituents that are tested separately (e.g. interpolation, halo swapping, and Fourier transforms). For a coded operator, $\mathbf{A}$, with input v_{in} and its coded

adjoint $\mathbf{A}^{\mathrm{T}}$, the adjoint test computes the left- and right-hand sides of the following formula, which must agree to machine precision to gain confidence that the coded adjoint is correct:

$$(\mathbf{A} v_{\mathrm{in}})^{\mathrm{T}} \mathbf{A} v_{\mathrm{in}} \overset{?}{=} v_{\mathrm{in}}^{\mathrm{T}} \mathbf{A}^{\mathrm{T}} \mathbf{A} v_{\mathrm{in}}, \tag{12}$$

where a random vector v_{in} will normally suffice. The CVT needs to be inverted in the gradient algorithm when using more than one outer loop (step 2a in Sect. 3.5) and in calibrating the $\mathbf{B}$ matrix (Sect. 4.3). These operators are subject to an *inverse test* to demonstrate that the inverse has been coded correctly. This is done by reading in a perturbation state and then passing it through $\mathbf{A} \mathbf{A}^{-1}$. The result is output, which can be compared to the original field read-in data. A test that the gradient of the cost function (as computed for the minimisation) is valid can be confirmed in a gradient test, which is also provided as part of the test suite. The gradient test estimates progressively more accurate finite-difference approximations to the gradient, and it checks that they converge to the analytically computed gradient (Eq. 11). Other tests are possible that have not been included in this version of ABC-DA, e.g. checks that the innovation statistics, namely $\left\langle (y - y^{\mathrm{m}}) (y - y^{\mathrm{m}})^{\mathrm{T}} \right\rangle$ equals to $\mathbf{R} + \mathbf{H} \mathbf{B}^{\mathrm{ic}} \mathbf{H}^{\mathrm{T}}$ (where $y^{\mathrm{m}}(t)$ is the background's version of the observations and the angled brackets indicate average over a large number of DA cycles with the same observation network).

4 Scientific and technical configuration of ABC-DA v1.4

This section is a description of the current scientific configuration of the ABC-DA system. This section also contains some technical information and can be read in conjunction with the user documentation available on GitHub (https://github.com/rossbannister/ABC-DA_1.4da/ blob/master/docs/Documentation.pdf, last access: 24 August 2020) where more information is available, including names of the executables to be run, the namelist variables that have to be set, and the input and output file names. References are made to this document in the sections below in the form of *GitHubDoc§x*. The code is divided into master programs that perform specific tasks. The relevant variables are set in a namelist file (filenames, options, switches, and parameters), and then the relevant executable is run. A list of the available master routines is listed in *GitHubDoc§2*, and instructions on how to download and build the code are found in *GitHubDoc§3*.

4.1 Construction of a model state and making a forecast

The initial conditions for a model run may be generated using the program *Master_PrepareABC_InitState* (*GitHubDoc§4.1*). The code can take a slice from a specific Met Office Unified Model (UM) file, or it can generate a simple

idealised pressure "blob" of specified position and size (or a combination of these). When initial fields are taken from the UM, they need to be adjusted to make them compatible with the ABC model. This involves a number of steps: (i) adjusting the fields towards the E and W edges of the domain to be consistent with the periodic boundary conditions; (ii) adding a constant to the v field to force its integral over each level to zero to allow GB in Eq. (2a) (balance condition, Eq. 2b, though, is not enforced to allow for some imbalance); (iii) computing $\tilde{\rho}'$ from v with Eq. (2a) and then adding a constant to force its integral over each level to zero; (iv) computing b' to satisfy the HB condition (Eq. 3); and (v) setting w' so that the 3D winds have zero divergence. A forecast can be made from these initial conditions using the program *Master_RunNLModel* (*GitHubDoc§4.2*) by numerically integrating Eq. (1).

4.2   The CVTs (B matrix) implemented

ABC-DA has a variety of options implemented to model the **B** matrix using control variable transforms (CVTs, Sect. 3.4), and this section describes the current implementation. The transforms are most easily understood by describing first the inverse CVTs (since they allow the difference "spaces" to be defined starting in model space and working towards the control space). The CVT operators defined are used in many of the programs mentioned in later sections.

4.2.1   The inverse parameter transform, U_p^{-1}

Recall that U_p^{-1} transforms a perturbation in model variables Eq. (1) to alternative parameters that are assumed to be uncorrelated. It is needed primarily to calibrate the CVT (Sect. 4.3). The input fields in this procedure are the perturbations δu, δv, δw, $\delta\tilde{\rho}'$, and $\delta b'$; the output fields are (in the version of the code documented) $\delta\psi$ (streamfunction), $\delta\chi_{vp}$ (velocity potential[1]), $\delta\tilde{\rho}'^u$ (unbalanced scaled density), $\delta b'^u$ (unbalanced buoyancy), and δw^u (unbalanced vertical wind). All input and output fields are a function of longitude and height. This is the algorithm for U_p^{-1}.

1. Compute the streamfunction:

$$\delta\psi = \nabla_x^{-1}\delta v. \tag{13}$$

   The operator ∇_x^{-1} is defined as $\nabla_x^{-1}\delta v = (\partial/\partial x)^{-2}\partial(\delta v)/\partial x$, which is based on application of the Helmholtz theorem (see Petrie et al., 2017, Sect. 4.1).

2. Compute the velocity potential (again based on the Helmholtz theorem):

$$\delta\chi_{vp} = \nabla_x^{-1}\delta u. \tag{14}$$

----
[1]Do not confuse the velocity potential perturbation, $\delta\chi_{vp}$, with the control vector, $\delta\chi$.

Note that in Eq. (13) $\delta\psi$ depends only on the meridional wind and in Eq. (14) $\delta\chi_{vp}$ depends only on the zonal wind. This is unlike a system that has latitude dependence, where $\delta\psi$ and $\delta\chi_{vp}$ would each depend on both δu and δv as per the Helmholtz theorem.

3. Compute the GB scaled density:

$$\delta\tilde{\rho}'^b = \alpha f\delta\psi/C, \tag{15}$$

   which follows from application of the Helmholtz theorem for this system, $(\delta u, \delta v) = (\partial\delta\chi_{vp}/\partial x, \partial\delta\psi/\partial x)$, applied to the GB equation (Eq. 2a). The value $\alpha = 1$, unless the system is configured to turn off GB in this transform, in which case $\alpha = 0$.

4. Compute the balanced scaled density after it has been vertically regressed:

$$\delta\tilde{\rho}'^{br} = \mathbf{R}_\rho\delta\tilde{\rho}'^b. \tag{16}$$

   The vertical regression (VR) operator $\mathbf{R}_\rho$ has the form $\mathbf{R}_\rho = \mathbf{C}^{\delta\tilde{\rho}'\delta\tilde{\rho}'^b}\left(\mathbf{C}^{\delta\tilde{\rho}'^b\delta\tilde{\rho}'^b}\right)^{-1}$, where $\mathbf{C}^{\delta\tilde{\rho}'^b\delta\tilde{\rho}'^b}$ is the correlation matrix between a previously computed population of $\delta\tilde{\rho}'^b$ perturbations with itself, and $\mathbf{C}^{\delta\tilde{\rho}'\delta\tilde{\rho}'^b}$ is the correlation matrix between $\delta\tilde{\rho}'^b$ and $\delta\tilde{\rho}'$. The justification for the use of $\mathbf{R}_\rho$ is given in Appendix B. The system can be configured to turn off this step (and is not used anyway if $\alpha = 0$ in step 3).

5. Compute the unbalanced scaled density:

$$\delta\tilde{\rho}'^u = \delta\tilde{\rho}' - \delta\tilde{\rho}'^{br} \tag{17}$$

   $(\delta\tilde{\rho}'^u = \delta\tilde{\rho}'$ if $\alpha = 0)$.

6. Compute the HB buoyancy:

$$\delta b'^b = \beta\mathbf{L}^{hb}\delta\tilde{\rho}'. \tag{18}$$

   The operator $\mathbf{L}^{hb}$ is defined as $\mathbf{L}^{hb}\delta\tilde{\rho}' = C\partial\tilde{\rho}'/\partial z$, as in Eq. (3). The value $\beta = 1$, unless the system is configured to turn off HB, in which case $\beta = 0$.

7. Compute the unbalanced buoyancy:

$$\delta b'^u = \delta b' - \delta b'^b \tag{19}$$

   $(\delta b'^u = \delta b'$ if $\beta = 0)$.

8. Compute the anelastically balanced vertical wind:

$$\delta w^b = \gamma\left(\mathbf{L}_u^{ab}\delta u + \mathbf{L}_{\tilde{\rho}'}^{ab}\delta\tilde{\rho}'\right). \tag{20}$$

   Using Eq. (1d), the operators $\mathbf{L}_u^{ab}$ and $\mathbf{L}_{\tilde{\rho}'}^{ab}$ are defined as $\mathbf{L}_u^{ab}\delta u = -(1/\tilde{\rho}_0)\int dz'\,\partial(\tilde{\rho}_0\delta u)/\partial x$ and $\mathbf{L}_{\tilde{\rho}'}^{ab}\delta\tilde{\rho}' =$

$-(1/\tilde{\rho}_0) \int dz' \left[\partial(u_0 \delta\tilde{\rho}')/\partial x + \partial(w_0 \delta\tilde{\rho}')/\partial z' \right]$ (integrating from the ground to height z), and $\delta\boldsymbol{w}^{\mathrm{b}}$ is the component of the vertical wind that, with $\delta\boldsymbol{u}$, has zero 3D divergence (sometimes called *anelastic balance* (AB); see Sect. 3.1.1 of Pielke, 2002). The value $\gamma = 1$, unless the system is configured to turn off AB, in which case $\gamma = 0$.

9. Compute the unbalanced vertical wind:

$$\delta\boldsymbol{w}^{\mathrm{u}} = \delta\boldsymbol{w} - \delta\boldsymbol{w}^{\mathrm{b}} \qquad (21)$$

$(\delta\boldsymbol{w}^{\mathrm{u}} = \delta\boldsymbol{w}$ if $\gamma = 0$).

Some of these steps may be omitted according to user options, as specified above. The above steps may be written more compactly as the following "super matrix":

$$\delta\boldsymbol{\chi} = \mathbf{U}_{\mathrm{p}}^{-1} \delta\boldsymbol{x},$$

$$
\begin{pmatrix} \delta\boldsymbol{\psi} \\ \delta\boldsymbol{\chi}_{\mathrm{vp}} \\ \delta\tilde{\boldsymbol{\rho}}'^{\mathrm{u}} \\ \delta\boldsymbol{b}'^{\mathrm{u}} \\ \delta\boldsymbol{w}^{\mathrm{u}} \end{pmatrix} =
$$

$$
\begin{pmatrix} \mathbf{0} & \nabla_x^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \nabla_x^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\alpha \mathbf{R}_\rho \frac{f}{C} \nabla_x^{-1} & \mathbf{0} & 1 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & -\beta \mathbf{L}^{\mathrm{hb}} & 1 \\ -\gamma \mathbf{L}_u^{\mathrm{ab}} & \mathbf{0} & 1 & -\gamma \mathbf{L}_{\tilde{\rho}'}^{\mathrm{ab}} & \mathbf{0} \end{pmatrix}
\begin{pmatrix} \delta\boldsymbol{u} \\ \delta\boldsymbol{v} \\ \delta\boldsymbol{w} \\ \delta\tilde{\boldsymbol{\rho}}' \\ \delta\boldsymbol{b}' \end{pmatrix}.
$$

$$(22)$$

It is noted here that this particular form of transform is not *necessarily* the most appropriate form for convective-scale systems, e.g. GB in step 3 and HB in step 6 *may* not be relevant. There is, however, expected to be some GB at the larger scales represented and HB at even shorter scales. Furthermore these relationships are still used in some operational systems, so their inclusion in this study is justified. The use of other balance relationships is possible, including statistical balance relationships (e.g. Derber and Bouttier, 1999; Chen et al., 2013; Bannister et al., 2020). An alternative balance relationship that may be applicable at convective scale is mentioned in the summary.

4.2.2 The forward parameter transform, $\mathbf{U}_{\mathrm{p}}$

$\mathbf{U}_{\mathrm{p}}$ transforms perturbations of parameters to model space. This transform (and its adjoint) is used at each iteration of the Var algorithm. The input fields in this procedure are the parameter field perturbations $\delta\boldsymbol{\psi}$, $\delta\boldsymbol{\chi}_{\mathrm{vp}}$, $\delta\tilde{\boldsymbol{\rho}}'^{\mathrm{u}}$, $\delta\boldsymbol{b}'^{\mathrm{u}}$, and $\delta\boldsymbol{w}^{\mathrm{u}}$; the output fields are $\delta\boldsymbol{u}$, $\delta\boldsymbol{v}$, $\delta\boldsymbol{w}$, $\delta\tilde{\boldsymbol{\rho}}'$, and $\delta\boldsymbol{b}'$. This is the algorithm for $\mathbf{U}_{\mathrm{p}}$.

1. Compute the zonal wind based on the Helmholtz theorem:

$$\delta\boldsymbol{u} = \nabla_x \delta\boldsymbol{\chi}_{\mathrm{vp}}. \qquad (23)$$

2. Compute the meridional wind (also based on the Helmholtz theorem):

$$\delta\boldsymbol{v} = \nabla_x \delta\boldsymbol{\psi}. \qquad (24)$$

3. Compute the balanced scaled density $\tilde{\boldsymbol{\rho}}'^{\mathrm{b}}$ (Eq. 15).

4. Compute the vertically regressed balanced scaled density $\delta\tilde{\boldsymbol{\rho}}'^{\mathrm{br}}$ (Eq. 16).

5. Compute the total scaled density:

$$\delta\tilde{\boldsymbol{\rho}}' = \delta\tilde{\boldsymbol{\rho}}'^{\mathrm{br}} + \delta\tilde{\boldsymbol{\rho}}'^{\mathrm{u}}. \qquad (25)$$

6. Compute the hydrostatically balanced buoyancy $\delta\boldsymbol{b}'^{\mathrm{b}}$ (Eq. 18).

7. Compute the total buoyancy:

$$\delta\boldsymbol{b}' = \delta\boldsymbol{b}'^{\mathrm{b}} + \delta\boldsymbol{b}'^{\mathrm{u}}. $$

8. Compute the anelastically balanced vertical wind $\delta\boldsymbol{w}^{\mathrm{b}}$ (Eq. 20).

9. Compute the total vertical wind:

$$\delta\boldsymbol{w} = \delta\boldsymbol{w}^{\mathrm{b}} + \delta\boldsymbol{w}^{\mathrm{u}}. \qquad (26)$$

Again, some of these steps may be omitted according to user options, as set out in Sect. 4.2.1. The above steps may be written more compactly as the super matrix:

$$\delta\boldsymbol{x} = \mathbf{U}_{\mathrm{p}} \delta\boldsymbol{\chi},$$

$$
\begin{pmatrix} \delta\boldsymbol{u} \\ \delta\boldsymbol{v} \\ \delta\boldsymbol{w} \\ \delta\tilde{\boldsymbol{\rho}}' \\ \delta\boldsymbol{b}' \end{pmatrix} =
$$

$$
\begin{pmatrix} \mathbf{0} & \nabla_x & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \nabla_x & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \alpha\gamma \mathbf{L}_{\tilde{\rho}'}^{\mathrm{ab}} \mathbf{R}_\rho \frac{f}{C} & \gamma \mathbf{L}_u^{\mathrm{ab}} \nabla_x & \gamma \mathbf{L}_{\tilde{\rho}'}^{\mathrm{ab}} & \mathbf{0} & 1 \\ \alpha \mathbf{R}_\rho \frac{f}{C} & \mathbf{0} & 1 & \mathbf{0} & \mathbf{0} \\ \alpha\beta \mathbf{L}^{\mathrm{hb}} \mathbf{R}_\rho \frac{f}{C} & \mathbf{0} & \beta \mathbf{L}^{\mathrm{hb}} & 1 & \mathbf{0} \end{pmatrix}
\begin{pmatrix} \delta\boldsymbol{\psi} \\ \delta\boldsymbol{\chi}_{\mathrm{vp}} \\ \delta\tilde{\boldsymbol{\rho}}'^{\mathrm{u}} \\ \delta\boldsymbol{b}'^{\mathrm{u}} \\ \delta\boldsymbol{w}^{\mathrm{u}} \end{pmatrix}. \quad (27)
$$

Using Eqs. (22) and (27), it may be confirmed that $\mathbf{U}_{\mathrm{p}} \mathbf{U}_{\mathrm{p}}^{-1} = \mathbf{I}$. The adjoint of Eq. (27) is constructed directly from the code.

4.2.3 The inverse spatial transform, $\mathbf{U}_{\mathrm{s}}^{-1}$

In the current configuration, the spatial transform comprises separate horizontal ($\mathbf{U}_{\mathrm{h}}$), vertical ($\mathbf{U}_{\mathrm{v}}$), and scaling ($\boldsymbol{\Sigma}$) transforms. The order of these transforms may vary. The first ordering is called the "classic transform order" (CTO, since

this was the transform order in the first Met Office Var system; Wlasak and Cullen, 2014),

$$\mathbf{U}_s^{-1} = \mathbf{U}_h^{-1}\mathbf{U}_v^{-1}\boldsymbol{\Sigma}^{-1}, \tag{28}$$

and the second is called the "reversed transform order" (RTO)

$$\mathbf{U}_s^{-1} = \mathbf{U}_v^{-1}\mathbf{U}_h^{-1}\boldsymbol{\Sigma}^{-1}. \tag{29}$$

$\boldsymbol{\Sigma}$ is a diagonal matrix of background error standard deviations of the parameters, as a function of longitude and height (although options are implemented to allow the standard deviation to be a function of height only or a constant for each parameter). After the parameters have been divided by $\boldsymbol{\Sigma}$, the problem remains one of modelling the covariances between spatial points in space.

There are separate spatial operators for each parameter defined in Sect. 4.2.1, and so strictly we should define the overall spatial transforms as block-diagonal forms, but we instead adopt a casual way of describing the transforms to avoid getting bogged down in notation. Depending on the context, these transforms may represent all parameters at once (as done in Sect. 3.4), single parameters or to individual horizontal levels or vertical columns (as is done below).

$\mathbf{U}_h$ and $\mathbf{U}_v$ have the same generic form, as follows:

$$\mathbf{U}_{h/v}^{-1} = \mathbf{G}\boldsymbol{\Lambda}^{-1/2}\mathbf{F}^{\mathrm{T}}, \tag{30a}$$

$$\text{so } \mathbf{U}_{h/v} = \mathbf{F}\boldsymbol{\Lambda}^{1/2}\mathbf{G}^{\mathrm{T}}, \tag{30b}$$

where $\mathbf{F}$ is the (exact or assumed) matrix of eigenvectors (columns of $\mathbf{F}$) of the covariance matrix that is being modelled, $\boldsymbol{\Lambda}^{1/2}$ is the diagonal matrix of eigenvalues, and $\mathbf{G}$ is any orthonormal square matrix ($\mathbf{G}^{\mathrm{T}}\mathbf{G} = \mathbf{I}$) of the same dimensions as $\boldsymbol{\Lambda}^{1/2}$. In Eq. (30a), $\mathbf{F}^{\mathrm{T}}$ projects a state onto the eigenvectors (mutually uncorrelated by definition), $\boldsymbol{\Lambda}^{-1/2}$ scales the projections so they have unit variance, and $\mathbf{G}$ is an arbitrary rotation. If the complete CVT had this form (but also incorporating $\boldsymbol{\Sigma}$), then the implied covariance would, by Eq. (10), be

$$\boldsymbol{\Sigma}\mathbf{F}\boldsymbol{\Lambda}^{1/2}\mathbf{G}^{\mathrm{T}}\left(\boldsymbol{\Sigma}\mathbf{F}\boldsymbol{\Lambda}^{1/2}\mathbf{G}^{\mathrm{T}}\right)^{\mathrm{T}} =$$
$$\boldsymbol{\Sigma}\mathbf{F}\boldsymbol{\Lambda}^{1/2}\mathbf{G}^{\mathrm{T}}\mathbf{G}\boldsymbol{\Lambda}^{1/2}\mathbf{F}^{\mathrm{T}}\boldsymbol{\Sigma} = \boldsymbol{\Sigma}\mathbf{F}\boldsymbol{\Lambda}\mathbf{F}^{\mathrm{T}}\boldsymbol{\Sigma},$$

where $\mathbf{F}\boldsymbol{\Lambda}\mathbf{F}^{\mathrm{T}}$ is the eigenvalue decomposition of the covariance matrix in question. In this illustration, the CVT is an exact representation of the covariances, but when this procedure is applied in practice, it is only an approximate covariance model, e.g. due to the separation of the horizontal and vertical transform or to the application of approximate eigenvectors. The structure of the actual implied covariances can be investigated with the software suite (Sect. 4.4).

In the ABC-DA system, we use the following for $\mathbf{F}$ and $\mathbf{G}$ in Eq. (5).

- For the horizontal transform $\mathbf{U}_h^{-1}$, $\mathbf{F}$ is called $\mathbf{F}_h$, whose columns comprise horizontal plane waves of the form $\sim \exp ikx$, where $i = \sqrt{-1}$ and k is the wavenumber (each column of $\mathbf{F}_h$ is a different k). In this context, $\mathbf{F}_h^{\mathrm{T}}$ represents a horizontal Fourier transform. This makes the assumption that the eigenvectors of the horizontal covariance matrix are plane waves, and the eigenvalues in $\boldsymbol{\Lambda}_h$ are their variances. This is equivalent to assuming horizontal error covariances that are homogeneous (see Bartello and Mitchell, 1992; Berre, 2000; Bannister, 2008b). $\mathbf{G}$ is set to $\mathbf{I}$ in the horizontal transform.

- For the vertical transform $\mathbf{U}_v^{-1}$, $\mathbf{F}$ is called $\mathbf{F}_v$ whose columns are the eigenmodes of a vertical covariance matrix (labelled with ν; see below), and $\boldsymbol{\Lambda}_v$ represents their variances. $\mathbf{G}$ is set either to $\mathbf{F}_v$ to give a symmetric vertical transform or to $\mathbf{I}$, depending on user choice.

The spaces that these operators work in depends on the chosen order of the transforms and on whether the vertical transform is symmetric or not. The following summarises these options, is repeated for each parameter, and can be read with Fig. 1, which shows how the transforms change the horizontal and vertical co-ordinates.

- For the CTO, $\mathbf{U}_v^{-1}$ operates vertically on a field that is a function of x and z. The vertical eigenvectors or values are those of a pre-computed horizontally averaged vertical covariance matrix, and so these matrices themselves are not dependent on horizontal position in ABC-DA.

  - For the symmetric vertical transform option, $\mathbf{U}_v^{-1} = \mathbf{F}_v\boldsymbol{\Lambda}_v^{-1/2}\mathbf{F}_v^{\mathrm{T}}$; the output of $\mathbf{U}_v^{-1}$ is also a field that is a function of x and z. The horizontal transform, $\mathbf{U}_h^{-1}$, then operates horizontally on such a field. The horizontal eigenvalues are those of pre-computed horizontal covariance matrices (one for each z in ABC-DA). The output of $\mathbf{U}_h^{-1}$ is a field that is a function of k and z (see Fig. 1a). This combination of options allows a different horizontal covariance to be specified for each vertical level.

  - For the non-symmetric vertical transform option, $\mathbf{U}_v^{-1} = \boldsymbol{\Lambda}_v^{-1/2}\mathbf{F}_v^{\mathrm{T}}$; the output of $\mathbf{U}_v^{-1}$ is a field that is a function of x and vertical eigenmode index ν. The horizontal transform, $\mathbf{U}_h^{-1}$, then operates horizontally on such a field. The horizontal eigenvalues are those of pre-computed horizontal covariance matrices (one for each ν in ABC-DA). The output of $\mathbf{U}_h^{-1}$ is a field that is a function of k and ν (see Fig. 1b). This combination of options allows a different horizontal covariance to be specified for each vertical mode, effectively allowing horizontal and vertical length scales to be associated.

- For the RTO, $\mathbf{U}_h^{-1}$ operates horizontally on a field that is a function of x and z. The horizontal eigenvalues are

(a) Classic transform order, symmetric vertical transform

(b) Classic transform order, non-symmetric vertical transform

(c) Reversed transform order, symmetric vertical transform

(d) Reversed transform order, non-symmetric vertical transform
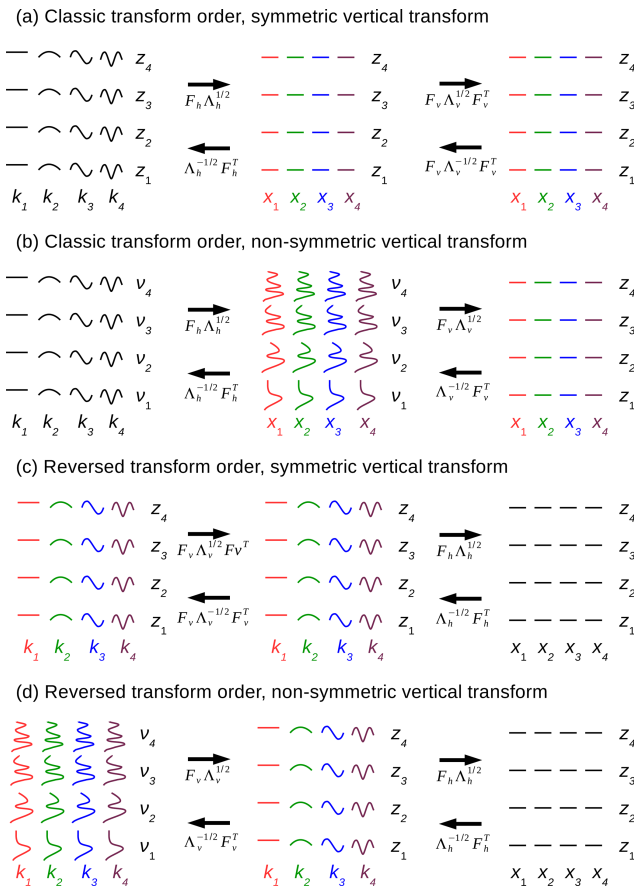

Figure 1. Schema to illustrate the different options for the spatial transforms as indicated by the panel titles (these are combinations of classic and reversed transform orders and symmetric and non-symmetric vertical transforms). Representations of the vertical direction include model levels (labelled with z_1, z_2, etc.) and vertical modes (v_1, v_2, etc.). Representations of the horizontal direction include model grid points (x_1, x_2, etc.) and Fourier modes (k_1, k_2, etc.). Moving from left to right indicates the forward transform and from right to left indicates the inverse transform. The horizontal transform is always done independently for each vertical coordinate (z_i to z_i or v_i to v_i), and the vertical transform is always done independently for each horizontal co-ordinate (x_i to x_i or k_i to k_i, as guided by the colours). The co-ordinates used in the control space in each option are indicated on the leftmost panels.

those of pre-computed horizontal covariance matrices (one for each z in ABC-DA). The output of $\mathbf{U}_h^{-1}$ is a field that is a function of k and z. The vertical transform, $\mathbf{U}_v^{-1}$, uses a separate set of vertical eigenvectors/values for each k.

- For the symmetric vertical transform option, the output of $\mathbf{U}_v^{-1}$ is also a field that is a function of k and z (see Fig. 1c).

- For the non-symmetric vertical transform option, the output of $\mathbf{U}_v^{-1}$ is a field that is a function of k and v (see Fig. 1d).

We would expect no difference between the implied covariances of the symmetric and non-symmetric vertical transform options in the reversed case; although, both options exist in the code.

4.2.4 The forward spatial transform, $\mathbf{U}_s$

The forward spatial transforms follow in a straightforward way from the inverses defined in Sect. 4.2.3, namely for the CTO

$$\mathbf{U}_s = \mathbf{\Sigma}\mathbf{U}_v\mathbf{U}_h, \tag{31}$$

and for the RTO

$$\mathbf{U}_s = \mathbf{\Sigma}\mathbf{U}_h\mathbf{U}_v. \tag{32}$$

The adjoint operators follow in a straightforward manner.

4.3 Calibrating the CVTs (B matrix)

The CVTs comprise many sub-matrices that need to be determined in a calibration procedure. The operators to be determined are the regression operator $\mathbf{R}_\rho$ (part of the parameter transform mentioned in Sect. 4.2.1 and 4.2.2) and $\mathbf{\Sigma}$, $\mathbf{\Lambda}_h$, $\mathbf{F}_v$, and $\mathbf{\Lambda}_v$ (parts of the spatial transforms mentioned in Sect. 4.2.3 and 4.2.4). The number of pieces of information to be determined in this procedure is explored in Appendix A. These matrices are determined from model training data in five stages, all using the program *Master_Calibration* (*GitHubDoc§4.4*), and they are stored in a covariance file, which is produced by this routine. It is impossible to use a genuine sample of forecast errors to calibrate the **B** matrix, so instead we use ensembles of forecast perturbations, which are considered proxies of forecast error (Buehner, 2005; Pereira and Berre, 2006). The five calibration stages are described here, and example outputs are shown for a standard set-up (experiment GB+VR+ to be described in Sect. 5.1).

4.3.1 Generate a population of training data from UM fields

This is calibration run stage 1 (*Master_Calibration* is run with the namelist variable *CalibRunStage* set to 1). This takes data from one or more UM files (one or more ensembles of forecasts) and extracts multiple longitude and height slices from these files to construct an effective "super ensemble". These are each adjusted to make them compatible with the ABC model (as in Sect. 4.1) followed by a short forecast of specified length (in our examples 1 h). These procedures are intended to give the ensemble members properties of the ABC model rather than the Unified Model from which they came, although the degree to which this has been achieved is not demonstrated. The super ensemble is output from this stage. Also specified at this stage are the model parameters (in the example to be described $A = 0.02\,\mathrm{s}^{-1}$, $B = 0.01$, and
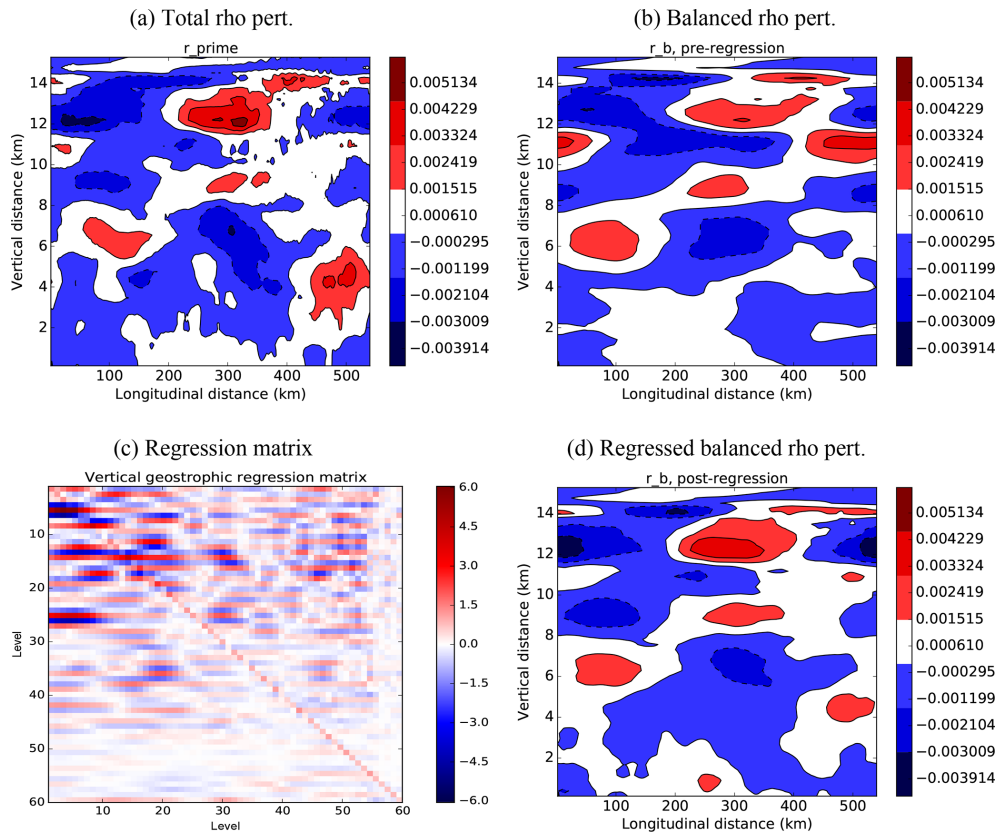
Figure 2. Plots comparing an example total scaled density perturbation, $\delta\tilde{\rho}'$, with the diagnosed balanced part, $\delta\tilde{\rho}'^{b}$, and showing the effect of the regression matrix. **(a)** $\delta\tilde{\rho}'$ (output from stage 2 of the calibration procedure), **(b)** $\delta\tilde{\rho}'^{b}$ (diagnosed from the streamfunction as in Eq. 15), **(c)** the regression matrix $\mathbf{R}_\rho$ (found from stage 3), and **(d)** its effect on $\delta\tilde{\rho}'^{b}$, i.e. $\delta\tilde{\rho}'^{br} = \mathbf{R}_\rho\delta\tilde{\rho}'^{b}$. Note that in panel **(c)** the lowermost level corresponds to the top of the matrix.

$C = 10\,000\,\mathrm{m^2\,s^{-2}}$) and user settings for the transform options mentioned in Sect. 4.2 (here, unless stated otherwise, the control options use GB ($\alpha = 1$), Sect. 4.2.1 step 3; VR, step 4; HB ($\beta = 1$), step 6; no anelastic balance ($\gamma = 0$), step 8; the CTO, Sect. 4.2.3; non-symmetric vertical transform, Sect. 4.2.3; and parameter standard deviations that are a function of vertical level only). These are all output in a provisional covariance file (netCDF format). At this stage, the file is blank apart from containing information on these options for future reference. These user options are read from this file in later stages of the calibration when the above mentioned matrices are computed and output to this covariance file. Technical information is given in *GitHubDoc§4.4.1*, and the ensemble members can be plotted using the Python program specified there.

In this paper, a super ensemble of 260 members is used. Appendix A shows that this is more than adequate to determine the spatial transform matrices and the vertical regression matrix.

4.3.2 Generate a population of forecast perturbations

This is calibration run stage 2 (*Master_Calibration* is run with the namelist variable *CalibRunStage* set to 2). The forecasts output from stage 1 are converted to means and perturbations from the means. See *GitHubDoc§4.4.2*, which includes plotting information.

4.3.3 Compute the vertical regression matrix $\mathbf{R}_\rho$

This is calibration run stage 3 (*Master_Calibration* is run with the namelist variable *CalibRunStage* set to 3). The perturbations from stage 2 are used to calculate populations of $\delta\tilde{\rho}'^{b}$ (Eq. 15). The vertical correlations between $\delta\tilde{\rho}'^{b}$ and itself and between $\delta\tilde{\rho}'$ and $\delta\tilde{\rho}'^{b}$ are then used to compute $\mathbf{R}_\rho$ in the way specified in point 4 of Sect. 4.2.1 (see also Appendix B). $\mathbf{R}_\rho$ is then output to the covariance file created in stage 1. See *GitHubDoc§4.4.3*, which includes plotting information.

Fig. 2a and b compare example $\delta\tilde{\rho}'$ and $\delta\tilde{\rho}'^{b}$ fields. The large-scale pattern of these fields is similar, with $\delta\tilde{\rho}'^{b}$ being smoother and of lower magnitude than $\delta\tilde{\rho}'$, indicating
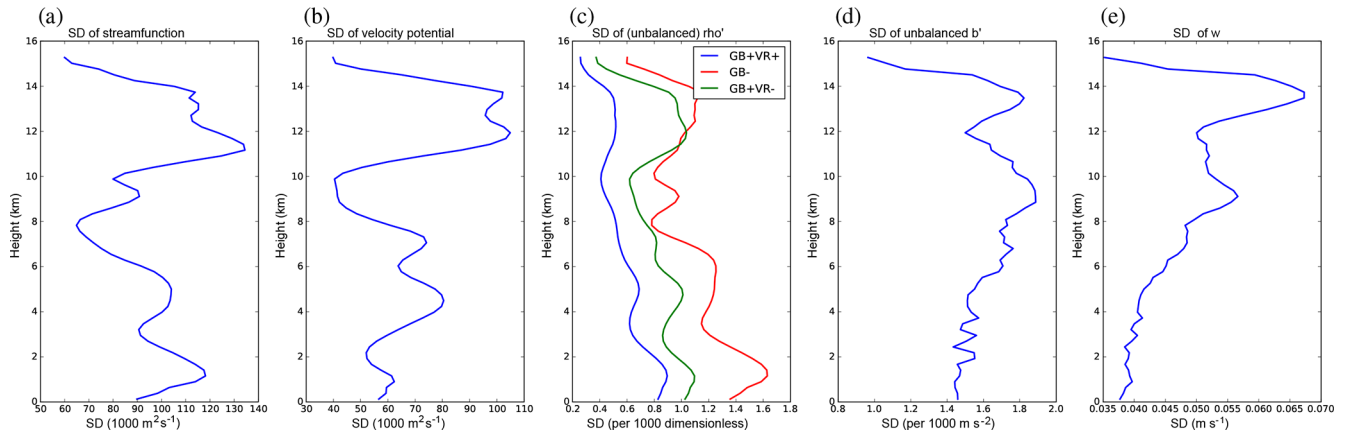
Figure 3. Profiles of background error standard deviations for the five control parameters with height: **(a)** streamfunction $\delta\psi$, **(b)** velocity potential $\delta\chi_{vp}$, **(c)** unbalanced scaled density $\delta\tilde{\rho}'^{u}$, **(d)** unbalanced buoyancy $\delta b'^{u}$, and **(e)** vertical wind δw. The blue lines are for the experiment described in the text – namely GB and VR are switched on in the parameter transform. In panel **(c)** the red line is for the experiment with GB (and hence VR) switched off, and the green line is for the experiment with GB switched on and VR switched off. In the other panels all experiments yield the same profiles. The values have been smoothed using a running average over the nearest five levels.

that the unbalanced contributions to $\delta\tilde{\rho}'$ are at smaller scales (as expected). Panel (c) shows an example $\mathbf{R}_\rho$ matrix and panel (d) shows its effect on $\delta\tilde{\rho}'^{b}$, showing its ability to modify values and vertical scales.

4.3.4 Perform the inverse parameter transform on the forecast perturbations

This is calibration run stage 4 (*Master_Calibration* is run with the namelist variable *CalibRunStage* set to 4). The perturbations from stage 2 are transformed to parameters using the procedure represented by $\mathbf{U}_p^{-1}$ (Sect. 4.2.1) and then output. The mean states found from stage 2 are also used in some of the calculations; e.g. $\tilde{\rho}_0$ is used in step 8 of that procedure. See *GitHubDoc§4.4.4*, which includes plotting information.

4.3.5 Calibrate the spatial transforms for each parameter

This is calibration run stage 5 (*Master_Calibration* is run with the namelist variable *CalibRunStage* set to 5). The perturbations from stage 4 are used to diagnose the matrices $\mathbf{\Sigma}$, $\mathbf{F}_v$, $\mathbf{\Lambda}_v$, and $\mathbf{\Lambda}_h$ for each of the five control parameters.

The blue lines of Fig. 3 show the background error standard deviations (diagonal elements of $\mathbf{\Sigma}$) for the control experiment described above (see the figure caption for a succinct summary). The parameters show some variability with height, although none of the parameters show variations of $\mathbf{\Sigma}$ of orders of magnitude. Note that the heights in the ABC model do not correspond with those in the real atmosphere.[2] The large variability of w (panel e) at model heights around

13 or 14 km for instance does not lie in the ABC model's stratosphere, and in any case without radiative forcing in the ABC model and an ozone layer, we would not expect signatures of the stratosphere to be present.

The blue lines in Fig. 4 show the square root of the eigenvalues of the vertical covariance matrix (diagonal elements of $\mathbf{\Lambda}_v^{1/2}$).[3] We find the vertical covariance matrices of each parameter over each super ensemble member and over each longitude. Each y axis in Fig. 4 is the (integer) vertical model index. The eigensolver sorts these into ascending value of eigenvalue, so the physical meaning of the modes can be unclear. For instance, examining the eigenvectors by eye (not shown), for parameters $\delta\psi$ (panel a), $\delta\chi_{vp}$ (panel b), and $\delta\tilde{\rho}'^{u}$ (panel c), the vertical modes of low index have vertical profiles that are generally rapidly oscillating and have more weight in the lower model levels than in the upper ones. As the vertical mode index increases, the oscillations, become less rapid and tend to have weight over the entire depth of the model atmosphere. There is no obvious trend concerning the vertical modes of $\delta b'^{u}$ (panel d) and δw (panel e). The values in Fig. 4 are of comparable magnitude between parameters because the vertical error covariance matrices are formed from the populations after they have been normalised with $\mathbf{\Sigma}^{-1}$ – see Eqs. (31) and (32).

Calibrating the vertical transform involves (for the CTO) constructing a single global vertical covariance matrix or (for the RTO) constructing one vertical covariance matrix for each wavenumber. The eigenvalues and eigenvectors follow from this procedure. The eigenvectors of the horizontal co-

---

[2]This is the case because, for simplicity, the irregularly spaced UM model levels are assigned new regularly spaced heights in the ABC model when generating training data in Sect. 4.3.1.

[3]In principle the vertical covariance matrix should actually be a correlation (rather than covariance) matrix given that the populations have been divided by $\mathbf{\Sigma}$. Due to the approximations made, the diagonal elements of this matrix may not be exactly unity.
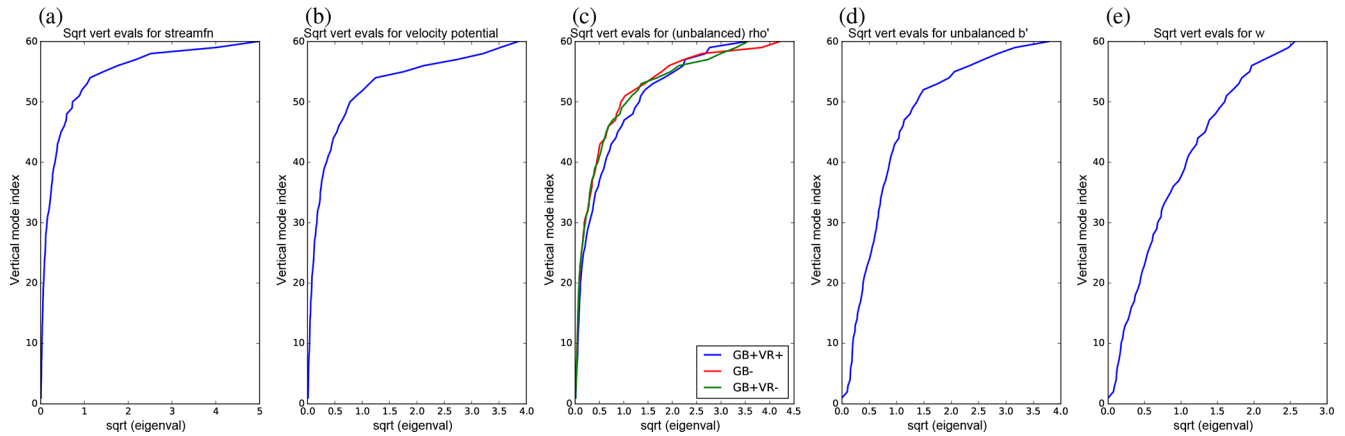
Figure 4. Profiles of the square root of the vertical eigenvalues for the five control parameters as a function of vertical mode index ν: **(a)** $\delta\psi$, **(b)** $\delta\chi_{vp}$, **(c)** $\delta\tilde{\rho}'^u$, **(d)** $\delta b'^u$, and **(e)** δw. The blue lines are for the experiment described in the text – namely GB and VR switched on. In panel **(c)** the red line is for the experiment with GB (and hence VR) switched off, and the green line is for the experiment with GB switched on and VR switched off. In the other panels, all experiments yield the same profiles.
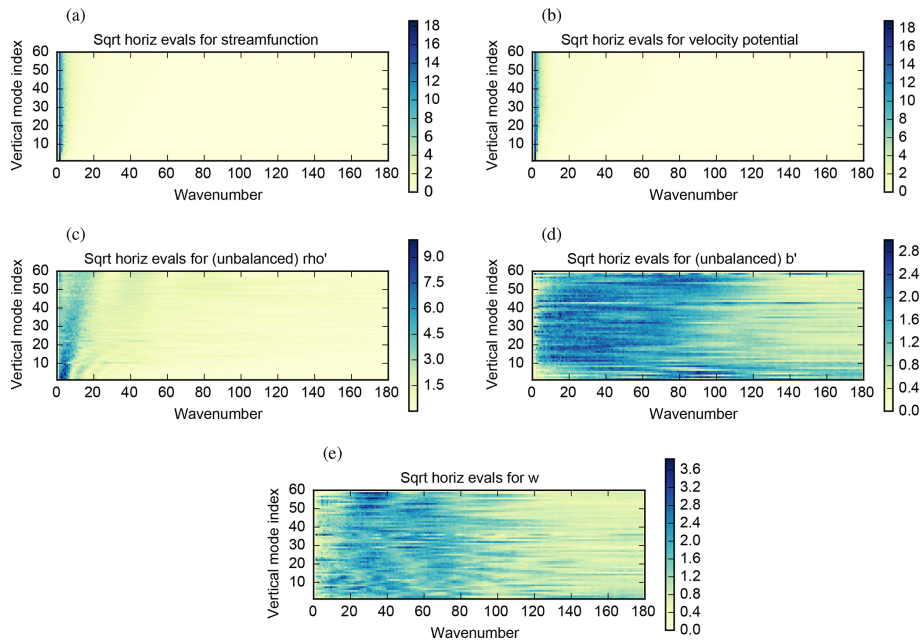
Figure 5. The square root of the horizontal variances (colours) as a function of horizontal wavenumber and vertical mode index for the five control parameters: **(a)** $\delta\psi$, **(b)** $\delta\chi_{vp}$, **(c)** $\delta\tilde{\rho}'^u$, **(d)** $\delta b'^u$, and **(e)** δw.

variance matrix are assumed to be plane waves. In the case of the CTO with a non-symmetric vertical covariance model, for example, the horizontal variances follow from passing each ensemble perturbation (δx) through the partial inverse transform to give $\delta\eta = \mathbf{F}_h^T \mathbf{U}_v^{-1} \mathbf{\Sigma}^{-1} \mathbf{U}_p^{-1} \delta x$. Here each $\delta\eta$ is a field which is a function of k and ν. This ensemble is then used to estimate the variances of $\delta\eta$ and $\mathbf{\Lambda}_h^0(k, \nu)$.

Figure 5 shows the square root of the horizontal variances (diagonal elements of $\mathbf{\Lambda}_h^{1/2}$). In the particular configuration that has a non-symmetric vertical transform, there is a differ-

ent horizontal variance spectrum for each vertical mode, so the axes in Fig. 5 are wavenumber, k, and vertical model index ν. The spectra for $\delta\psi$ (panel 5a) and $\delta\chi_{vp}$ (panel 5b) are similar and represent perturbations with long horizontal correlation length scales (most weight is at small wavenumbers). Their similarities are explained by the fact that the fields are related to the horizontal winds in the same way: $v = \partial\delta\psi/\partial x$ and $u = \partial\delta\chi_{vp}/\partial x$. The spectra for $\delta\tilde{\rho}'^u$ (panel 5c) show that perturbations in this parameter have shorter horizontal length scales than in panels (5a) and (5b), and the length scale re-
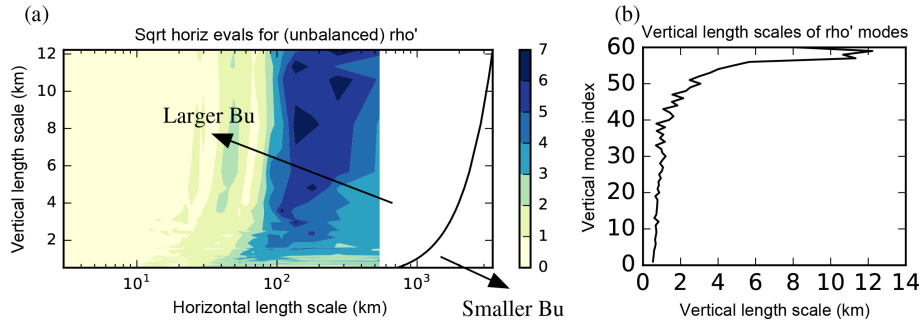
Figure 6. (a) The same horizontal spectra of Fig. 5c but shown as a function of horizontal (l_h) and vertical (l_v) length scales. Also shown in **(a)** is the Rossby radius of deformation, L_R, plotted as a function of l_v (black line). The Burger number is defined as $Bu = L_R/l_h$. **(b)** Vertical length scales (x axis) of the vertical modes (y axis) for the unbalanced scaled density control parameter, calculated according to Appendix C.

duces with higher vertical modes. This suggests that perturbations with broader vertical scales (generally higher vertical mode index) tend to have shorter horizontal scales. The spectra for the remaining parameters, $\delta b'^u$ (panel 5d) and δw (panel 5e), show that perturbations of these parameters are courser still, and there is no obvious pattern with vertical mode index.

The same data plotted in Fig. 5c (for $\delta\tilde{\rho}'^u$) are also plotted in Fig. 6a but with transformed co-ordinates to facilitate further interpretation. In Fig. 6a the variance square roots are shown as a function of horizontal length scale l_h and vertical length scale l_v; l_h is related to horizontal wavenumber index k (an integer), as $l_h = L_h/k$ (where $L_h = 540$ km is the horizontal domain length and excluding $k = 0$ since it gives an infinite l_h), and l_v is related to the vertical eigenmode index for $\delta\tilde{\rho}'^u$ in the way outlined in Appendix C. For reference, panel (5b) shows l_v for each vertical mode of $\delta\tilde{\rho}'^u$, which confirms that higher ν is broadly associated with deeper structures. Furthermore, modes with deeper structures have fewer nodes (not shown). Incidentally, this correspondence between ν and l_v is also found for the vertical modes of $\delta\psi$, which represent the balanced scaled density increments.

Returning to panel (a), the Rossby radius of deformation, L_R, has also been plotted (the black curve). This has been estimated using a shallow water interpretation; i.e. $L_R = \sqrt{gl_v}/f$ (e.g. Cullen, 2006). The Burger number, Bu, is defined as $Bu = L_R/l_h$, and it follows that all of the represented variability in these unbalanced modes has large Bu. Dynamical theory says that for $l_h \ll L_R$ (i.e. $Bu \gg 1$), the rotational wind is a potential vorticity (PV)-like variable; for $l_h \gg L_R$ ($Bu \ll 1$), the mass variable becomes a PV-like variable (e.g. Cullen, 2003; Katz et al., 2011). PV is often associated with balanced flow, so the above theory suggests that the unbalanced mass variable ($\delta\tilde{\rho}'^u$) should vanish – and hence have vanishing variance – for $Bu \ll 1$. Figure 6a shows that for 10^1 km $\lesssim l_h \lesssim 10^3$ km, the variance of $\delta\tilde{\rho}'^u$ does increase as Bu increases, showing some consistency with the dynamical theory at synoptic scales. It further suggests that a rotational

wind variable, like $\delta\psi$, is a reasonable variable to capture the balanced part of the flow at synoptic scales.

4.4 The implied B matrix

The covariance model is completely described by (i) the parameter transform presented in Sect. 4.2.2, (ii) the results presented in Sect. 4.3.5, and (iii) other statistics like the vertical eigenvectors themselves (not shown). The software suite has a program to compute the implied covariances (*Master_ImpliedCov*; see *GitHubDoc§4.6*), which uses information in the CVT file output by the calibration procedures. The program acts with the combination of matrices $\mathbf{UU}^T$ on a vector which is zero apart from one element (which we call the source point), which is set to unity. (In the example below, this element corresponds to $\delta\tilde{\rho}'$ near the centre of the domain.) Completing the matrix algebra, the implied super matrix (for fields δu, δv, δw, $\delta\tilde{\rho}'$, $\delta b'$) is found from Eq. (10); $\mathbf{U} = \mathbf{U}_p\mathbf{U}_s$, and Eq. (27):

$$\begin{pmatrix} \mathbf{B}_u^{ic} & 0 & \gamma\mathbf{B}_u^{ic}\mathbf{L}_u^{abT} & 0 & 0 \\ & \nabla_x\mathbf{B}_s^{\psi}\nabla_x^T & \alpha\gamma\frac{f}{C}\nabla_x\mathbf{B}_s^{\psi}\mathbf{R}_{\tilde{\rho}'}^T\mathbf{L}_{\tilde{\rho}'}^{abT} & \alpha\frac{f}{C}\nabla_x\mathbf{B}_s^{\psi_r}\mathbf{R}_{\tilde{\rho}'}^T & \alpha\beta\frac{f}{C}\nabla_x\mathbf{B}_s^{\psi}\mathbf{R}_{\tilde{\rho}'}^T\mathbf{L}^{hbT} \\ & & \mathbf{B}_w^{ic} & \alpha^2\gamma\frac{f^2}{C^2}\mathbf{L}_{\tilde{\rho}'}^{ab}\mathbf{B}_s^{\psi_r} + \gamma\mathbf{L}_u^{ab}\mathbf{B}_{\tilde{\rho}'}^{\tilde{\rho}'u} & \alpha^2\beta\gamma\frac{f}{C}\mathbf{L}_{\tilde{\rho}'}^{ab}\mathbf{B}_s^{\psi}\mathbf{L}^{hbT} + \gamma\beta\mathbf{L}_u^{ab}\mathbf{B}_{\tilde{\rho}'}^{\tilde{\rho}'u}L^{hbT} \\ & & & \alpha^2\frac{f^2}{C^2}\mathbf{B}_s^{\psi_r} + \mathbf{B}_{\tilde{\rho}'}^{\tilde{\rho}'u} & \alpha^2\beta\frac{f^2}{C^2}\mathbf{B}_s^{\psi_r}\mathbf{L}^{hbT} + \beta\mathbf{B}_{\tilde{\rho}'}^{\tilde{\rho}'u}\mathbf{L}^{hbT} \\ & & & & \mathbf{B}_{b'}^{ic} \end{pmatrix}.$$

$$(33)$$

where to save space only the upper triangular part of the covariance matrix is shown. (The rest of the matrix is the adjoint of the corresponding transpose element.) The following definitions have been made for convenience:

$$\mathbf{B}_s^p = \mathbf{U}_s^p\mathbf{U}_s^{pT}, \qquad p = \delta\psi, \delta\chi_{vp}, \delta\tilde{\rho}'^u, \delta b'^u, \text{ or } \delta w^u, \quad (34a)$$

$$\mathbf{B}_u^{ic} = \nabla_x\mathbf{B}_s^{\chi_{vp}}\nabla_x^T, \tag{34b}$$

$$\mathbf{B}_w^{ic} = \alpha^2\gamma^2\frac{f^2}{C^2}\mathbf{L}_{\tilde{\rho}'}^{ab}\mathbf{B}_s^{\psi_r}\mathbf{L}_{\tilde{\rho}'}^{abT} + \gamma^2\mathbf{L}_u^{ab}\mathbf{B}_u^{ic}\mathbf{L}_u^{abT}$$

$$+ \gamma^2\mathbf{L}_{\tilde{\rho}'}^{ab}\mathbf{B}_s^{\tilde{\rho}'u}\mathbf{L}_{\tilde{\rho}'}^{abT} + \mathbf{B}_s^{w^u}, \tag{34c}$$

$$\mathbf{B}_s^{\psi_r} = \mathbf{R}_\rho \mathbf{B}_s^\psi \mathbf{R}_\rho^T, \tag{34d}$$

$$\mathbf{B}_{b'}^{ic} = \alpha^2 \beta^2 \frac{f^2}{C^2} \mathbf{L}^{hb} \mathbf{B}_s^{\psi_r} \mathbf{L}^{hbT} + \beta^2 \mathbf{L}^{hb} \mathbf{B}_s^{\tilde{\rho}'u} \mathbf{L}^{hbT} + \mathbf{B}_s^{b'u}. \tag{34e}$$

Recall that the factors α, β, and γ are set to 1 or 0 to turn on or off geostrophic, hydrostatic, or anelastic balance in the covariance model (Sect. 4.2). Equation (34a) represents the implied spatial covariances, where p (parameter) is short for each control parameter listed in Eq. (34a). The diagonal blocks of Eq. (33) are the implied covariances of the model variables. The structure of the terms in Eq. (33) reveals the way that the covariances have been modelled. For example, the implied covariance of $\delta\tilde{\rho}'$ is the 4, 4th block, $\alpha^2 \frac{f^2}{C^2} \mathbf{B}_s^{\psi_r} + \mathbf{B}_s^{\tilde{\rho}'u}$, and comprises a balanced contribution (the first term, which depends upon the covariance structure of $\delta\psi$) and an unbalanced contribution (the second term, which is independent). Other variables have a similar decomposition.

This equation is too complicated to illustrate the implied geographical covariance structure, so a selection of the implied covariances (Eq. 10) are computed and shown graphically in the top row of Fig. 7 (Fig. 7a, b, c). All panels represent the covariances between a source point ($\delta\tilde{\rho}'$ at the position of the cross) with $\delta\tilde{\rho}'$ (panel 7a), δv (panel 7b), and $\delta b'$ (panel 7c). For comparison, the bottom row shows the respective raw sample covariances found from the population of 260 forecasts used to calibrate the covariance model (found using the program *Master_RawCov*; see *GitHubDoc§4.7*). We regard the signals contained in the raw covariances as a rough (row-rank) guide to the covariances that should ideally be modelled by the CVT.

The autocovariances in panel (7a) have a region of positive correlations about the source point, with regions of negative correlation towards the east, west, above, and below. This broad structure is apparent in the raw covariance panel (7j) although the implied values are 2 to 4 times larger. The covariances between $\delta\tilde{\rho}'$ and δv in panel (7b) show dipole patterns in the horizontal due to GB imposed according to Eq. (2a). This is done in the CVT via the 2, 4th block of Eq. (33), which spreads out the source point with $\mathbf{B}_s^\psi \mathbf{R}_\rho^T$, evaluates the horizontal gradient, and then multiplies by $\alpha\frac{f}{C}$. Physically, panel (b) is the meridional wind response (an anti-cyclone) to a positive density perturbation at the cross, and these covariances mean that this pattern would appear in the v analysis increments when assimilating a single $\tilde{\rho}'$ measurement at the cross. The pattern in panel (b) is evident in the raw covariance in panel (k), but the implied version is slightly too large. Finally, the covariances between $\delta\tilde{\rho}'$ and $\delta b'$ in panel (c) show the hydrostatic response of the density perturbation, which is the 5, 4th block of the CVT, $\alpha^2\beta\frac{f^2}{C^2}\mathbf{L}^{hb}\mathbf{B}_s^{\psi_r} + \beta\mathbf{L}^{hb}\mathbf{B}_s^{\tilde{\rho}'u} = \beta\mathbf{L}^{hb}\left(\alpha^2\frac{f^2}{C^2}\mathbf{B}_s^{\psi_r} + \mathbf{B}_s^{\tilde{\rho}'u}\right)$. There are two contributions to the

$\delta\tilde{\rho}'$–$\delta b'$ covariances: one involves the route $\delta\tilde{\rho} \rightarrow \delta\psi \rightarrow \delta b'$ (exploiting GB and HB) and the other involves the route $\delta\tilde{\rho} \rightarrow \delta\tilde{\rho}'^u \rightarrow \delta b'$ (exploiting only HB). A closer inspection of Eq. (33) reveals that the 5, 4th block is equal to $\beta\mathbf{L}^{hb}$ (proportional to the vertical derivative operator – see line after Eq. 18) acting on the 4, 4th block, which is the $\delta\tilde{\rho}'$ response. Figure 7 shows that panel (7c) is indeed related to the vertical derivative of panel (7a). The structure of panel (7c) comprises complicated bands of positive and negative covariances, and again this pattern would appear in the b' analysis increments when assimilating a single $\tilde{\rho}'$ measurement at the cross. The structure in panel (7c) is evident in the raw covariances panel (l), suggesting that HB is followed by the forecasts to some extent although the implied covariances are up to about 3 times larger.

4.5 Observations and observation operators

ABC-DA currently supports observation operators for direct (point) observations of u, v, w, $\tilde{\rho}'$, b', a conserved tracer, $\sqrt{u^2 + v^2}$, and $\sqrt{u^2 + v^2 + w^2}$. These correspond to observation codes 1–8, respectively. Observations are read in via a text file, which specifies the time of each observation, its longitudinal and height position, the observation code, the observation value itself, and its error standard deviation. The format of the input file is specified in *GitHubDoc§4.8*. The observation operators employ a bi-linear interpolation, and observation errors are assumed to be uncorrelated.

The user may extend the range of observation operators available by assigning a new observation code and adding a new branch in the observation operator and adjoint routines. Furthermore, observations are assigned a batch number. This is not used in the current code, but it may be used by the user to allow for correlated errors within each batch.

4.6 Generating data for twin experiments

The program *Master_MakeBgObs* (see *GitHubDoc§4.8*) is available to generate an observation network, to generate synthetic random observations given an observation network (for later assimilation), and to generate a random background state that is consistent with the background errors as specified in a CVT file. Further details are as follows.

– An observation network is a specification of a grid of observation times, locations, and error standard deviations, as well as which synthetic observations to make (but not the observation values themselves). For example, the user may wish to generate synthetic observations of $s = \sqrt{u^2 + v^2}$ in an $N_x^s \times N_z^s$ grid between x_1^s and x_2^s and between z_1^s and z_2^s. This is also true for synthetic observations of $\tilde{\rho}'$ in an $N_x^{\tilde{\rho}'} \times N_z^{\tilde{\rho}'}$ grid between $x_1^{\tilde{\rho}'}$ and $x_2^{\tilde{\rho}'}$ and between $z_1^{\tilde{\rho}'}$ and $z_2^{\tilde{\rho}'}$. Running *Master_MakeBgObs* with the option to generate this observation network (specified in an input file) will produce a
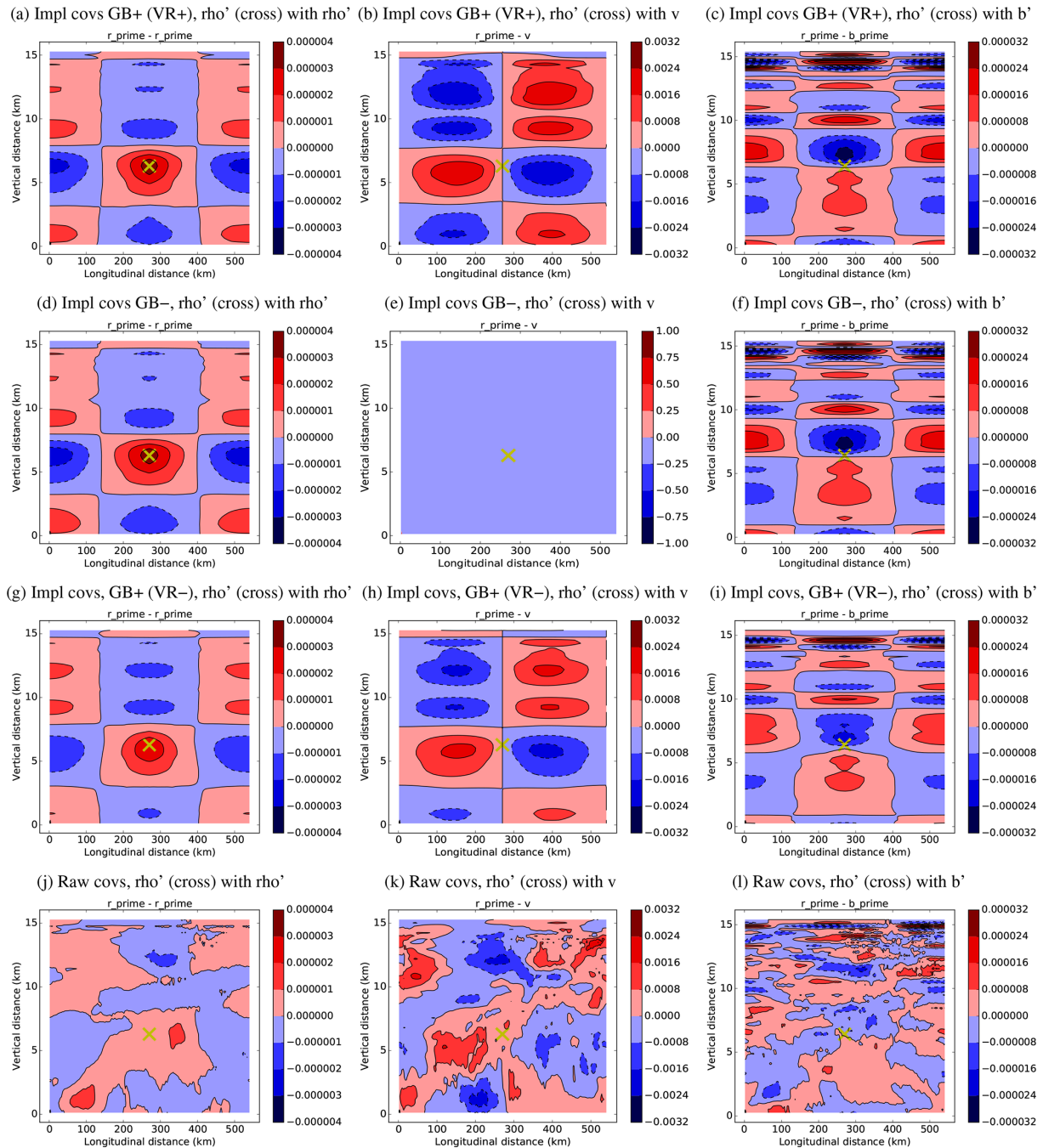
Figure 7. A selection of covariances (panel columns) found in different ways (rows). All covariances are between the source point ($\delta\tilde{\boldsymbol{\rho}}'$ at the cross) and $\delta\tilde{\boldsymbol{\rho}}'$ (left column), $\delta\boldsymbol{v}$ (middle column), and $\delta\boldsymbol{b}'$ (right column) at the field locations. The rows correspond to implied covariances when GB and VR are used in the CVT (GB+VR+, top row), implied covariances when GB is not used (GB−, second row), implied covariances when GB is used and VR is not used (GB+VR−, third row), and when covariances are calculated directly from the population used to calibrate each CVT (bottom row). Red (blue) indicates positive (negative) covariances.

text file containing entries for each individual observation (where it should be made, when it should be made, which quantity should be measured, and at what precision – see *GitHubDoc§4.8*). This file could be produced manually, but this software option exists to make the task less tedious. Once produced, this file can be edited by removing, adding, or altering individual observation points.

– Pseudo observations are generated according to the output from the above procedure and an ABC model initial condition (*truth*) file. Running *Master_MakeBgObs* in this mode will run a forecast from the initial conditions, generate "true observations" from the forecast, and will then add observation noise to each observation. The output from this stage is a file of synthetic observations which can then be assimilated (Sect. 4.7). The true observations are also output within this data file, which can be used later for error analysis. The user may wish to generate synthetic observations for an identical twin experiment by using the same model parameters (i.e. A, B, C) as will be used in the DA or may wish to run a non-identical twin experiment with different parameters. As with the procedure for generating the observation network, this file can be edited or instead created manually from scratch. For example, the user may want to add, remove, or change individual observations in order to investigate the effect on the assimilation.

– A valid background state may be generated with the program *Master_MakeBgObs*. The *truth* state (conventionally that used to generate the synthetic observations) is perturbed with a random state $\delta x = \mathbf{U}\delta\chi$ (Eq. 8), where $\delta\chi$ is drawn from $N(0, \mathbf{I})$. The information used to describe the CVT, $\mathbf{U}$, is found from a specified CVT file.

4.7 Data assimilation

The program that performs a DA cycle is called *Master_Assimilate*; see *GitHubDoc§4.9*. The inputs to this program are (i) a background state (e.g. as generated in Sect. 4.6), (ii) a set of observations (e.g. as generated in Sect. 4.6), and (iii) a description of the $\mathbf{B}$ matrix (in a CVT as found using the calibration procedure in Sect. 4.3). The assimilation methods currently available are 3DVar and 3DF-GAT, although the code is flexible enough to allow other methods (e.g. 4DVar, EnVar; Bannister, 2017) to be developed.

The algorithm for 3DFGAT (the preferred option in the current version) is shown in Fig. 8. The reference state, x^{r}, is set to the background, x^{b}, which is then used to compute the innovations, $d(t)$, computed at their appropriate times within the DA window. The control space perturbation, $\delta\chi$ is set to zero. $d(t)$ and $\delta\chi$ enter the inner-loop minimisation (pink box), where the residuals, $\mathbf{r}(t)$, are computed. These are used in the calculation of $\nabla_{\delta\chi}J_{\mathrm{o}}$, where J_{o} is the ob-
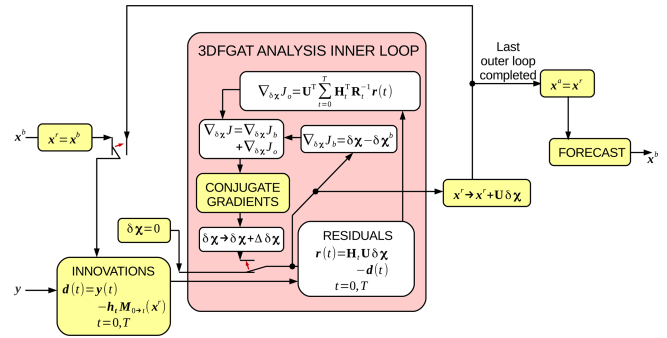

Figure 8. The outer and inner loops of the variational 3DFGAT procedure. The main inputs into the algorithm are the background state, x^{b}, and the observations, y. The algorithm inside the pink box is the inner loop. The small red arrows represent switches in the data flow. The initial positions (shown) are changed after the first iteration. This algorithm incorporates cycling, where the forecast made (on the right-hand side) is fed back (on the left-hand side) for the next assimilation window. The full 4DVar algorithm to calculate the gradient $\nabla_{\delta\chi}J$ (essentially part of the inner loop) is given in Sect. 3.5. Note that $\delta\chi^{\mathrm{b}} = \mathbf{U}^{-1}\left(x^{\mathrm{b}} - x^{\mathrm{r}}\right)$.

servation term of the cost function. This is added with the background contribution, $\nabla_{\delta\chi}J_{\mathrm{b}}$, giving the total cost function gradient. This is used with a descent algorithm – in our case a conjugate gradient algorithm – which provides the increment, $\Delta\delta\chi$. The switch is flipped (red arrow inside the pink box) so that the inner-loop procedure is repeated with the $\delta\chi$ that is updated with $\Delta\delta\chi$. Once converged, the reference state is incremented by $\mathbf{U}\delta\chi$. This ends the first outer-loop iteration. If further outer-loop iterations are required, the switch is flipped (red arrow outside of the pink box) so that the incremented x^{r} is fed into the innovation calculation. The x^{r} at the final outer loop is the analysis, x^{a}.

In addition to the analysis, and the analysis increment, there are many diagnostic outputs from the DA, as detailed in *GitHubDoc§4.9*. This includes the reference states at each outer loop, the gradient $\nabla_{\delta x}J_{\mathrm{o}}$, the values of the cost function with iteration, and versions of the observations file with extra information. The last file is output for each outer loop and includes the model observation values computed from x^{r} (the x^{r} corresponding to the last outer loop is x^{a}), the innovation, $d(t)$, the residual $\mathbf{r}(t)$, and the gradient of the observation term in the cost function with respect to the model observation, $\mathbf{R}_t^{-1}\mathbf{r}(t)$. These diagnostics can be plotted using the Python program specified in *GitHubDoc§4.9*. A selection of example diagnostics is shown in Sect. 5 in connection with the study on the effect of the GB conditions in the CVT.

4.8 Cycling

A (provided) Bash script controls the schedule of tasks to perform DA cycling; see *GitHubDoc§5*. The background state for the first cycle is made using the procedure in

Sect. 4.6, but subsequent backgrounds are found from forecasts originating from each analysis. The script also propagates the true state from one cycle to the next for generation of the synthetic observations and for subsequent monitoring.

Summary statistics of cycling experiments and comparative diagnostics between pairs of cycling experiments can be computed and plotted using Python programs provided. The study in Sect. 5 shows many of these diagnostics, which may be regarded as examples.

5   An investigation of balance in modelling the B matrix

Most Var systems began their lives serving global models on relatively coarse grids. In mid- and high-latitude regions, and when only relatively large scales are represented, a significant portion of the flow fields are related via GB and HBs (GB and HB). Most Var systems use this property to imply relationships between variables, which impact the background error covariance statistics, and a similar approach is often followed in systems that use much finer grids (as is done in Sect. 4.2 for ABC-DA). Since GB becomes a less important constraint at smaller scales (Berre, 2000; Bannister et al., 2011; Petrie et al., 2017), a research question arises concerning whether using such a constraint in the **B** model is necessary or indeed harmful in convective-scale DA.

Some regional Var systems, such as those for the AL-ADIN (Aire Limitée Adaption Dynamique et développement InterNational; Berre, 2000) and WRF (Weather Research and Forecasting model; Chen et al., 2013) models couple mass and wind fields via a linear regression operator, which is trained from sample data. The degree of balance represented by these operators will be appropriate to the training sample. Other systems, such as the current configuration of ABC-DA (Sect. 4.2), the Met Office (Ingleby, 2001; Bannister, 2008b), and HIRLAM (HIgh Resolution Limited Area Model; Gustafsson et al., 2001) impose analytic GB operators. Analytic relationships are "cleaner" and more physical than regressions, but their current forms are not so adaptive to the underlying conditions. Geophysical balances are not only sensitive to scale and latitude but also to precipitation, where heavy precipitation leads to a weakening of the balance between mass and wind (e.g. Caron and Fillion, 2010; Chen et al., 2016). Few, if any, studies have examined the effect of switching off the GB constraint completely in the **B** matrix in convective-scale systems. This is one way to affect the implied covariances ($\alpha = 1 \rightarrow 0$ in Eq. 33). Other suggestions have been made to change the nature of the wind coupling in **B** at convective scale. Sun et al. (2016), for example, compared using ψ, χ-based uncorrelated control parameters with u, v-based ones (with the associated change of the CVT). Their comparison of these two alternative sets of wind control parameters was not so straightforward as their ψ, χ-based set used a linear regression-type balance with mass variables, while their u, v-based set did not. This did not allow for a clean analysis of the effect of the change of parameters versus the effect of turning off the balance constraint.

5.1   Description of the DA experiments

In this section we do three DA experiments each with its own configuration as described below.

- Experiment GB+VR+ mirrors the operational configuration of the Met Office DA system. In this configuration $\delta\tilde{\boldsymbol{\rho}}'$ is decomposed into a GB part, $\delta\tilde{\boldsymbol{\rho}}'^{\mathrm{b}}$, and an unbalanced part, $\delta\tilde{\boldsymbol{\rho}}'^{\mathrm{u}}$ ($\alpha = 1$ in Eqs. 27 and 33), and it uses the VR operator, $\mathbf{R}_\rho$, to modify the calculation of the balanced scaled density perturbation increments from the streamfunction increment. $\delta\boldsymbol{b}'$ is decomposed into a HB part, $\delta\boldsymbol{b}'^{\mathrm{b}}$, and an unbalanced part, $\delta\boldsymbol{b}'^{\mathrm{u}}$ ($\beta = 1$). $\delta\boldsymbol{w}$ is analysed as though it were completely unbalanced ($\gamma = 0$). The ordering of the vertical and horizontal transforms is the CTO, and the non-symmetric vertical transform is used. As a reminder, the implied covariances for GB+VR+ are shown in the top row of Fig. 7.

- Experiment GB− differs from GB+VR+ in that there is no balanced component of $\delta\tilde{\boldsymbol{\rho}}'$ ($\alpha = 0$, $\beta = 1$, and $\gamma = 0$). Since there is no GB operator, the VR step is irrelevant. The implied covariances for GB− are shown in the second row of Fig. 7. When calibrated with this configuration, the $\delta\tilde{\boldsymbol{\rho}}' - \delta\tilde{\boldsymbol{\rho}}'$ covariances (panel d) are almost the same as for GB+VR+ (see the 4, 4th block of Eq. (33) to see how α affects the $\delta\tilde{\boldsymbol{\rho}}' - \delta\tilde{\boldsymbol{\rho}}'$ covariances). The $\delta\tilde{\boldsymbol{\rho}}' - \delta\boldsymbol{v}$ covariances (panel e) are zero as expected as GB was the mechanism used to couple these two variables in the CVT (see the 2, 4th block of Eq. 33). As $\delta\tilde{\boldsymbol{\rho}}'^{\mathrm{b}} = 0$ in this experiment, $\delta\tilde{\boldsymbol{\rho}}'^{\mathrm{u}} = \delta\tilde{\boldsymbol{\rho}}'$, which is consistent with the variances of $\delta\tilde{\boldsymbol{\rho}}'^{\mathrm{u}}$ (red line in Fig. 3c for GB−) being larger than those of the blue line (GB+VR+). The $\delta\tilde{\boldsymbol{\rho}}' - \delta\boldsymbol{b}'$ covariances in Fig. 7f are similar to those for GB+VR+ (panel c), showing that the effect of GB on these statistics is small.

- Experiment GB+VR− differs from GB+VR+ in that no VR is performed ($\alpha = 1$, $\beta = 1$, $\gamma = 0$, and $\mathbf{R}_\rho = \mathbf{I}$). The implied covariances for GB+VR− are shown in the third row of Fig. 7. When calibrated with this configuration, the $\delta\tilde{\boldsymbol{\rho}}' - \delta\tilde{\boldsymbol{\rho}}'$ covariances (panel g) are slightly smaller than for GB+VR+. The effect on $\delta\boldsymbol{v}$ (panel h) is noticeably smaller than in GB+VR+. In GB+VR+, $\mathbf{R}_\rho$ has the effect of increasing the magnitude of the balanced increment ($\delta\tilde{\rho}_{\mathrm{r}}'^{\mathrm{b}}$ is greater than $\delta\tilde{\boldsymbol{\rho}}'^{\mathrm{b}}$ on average) and thus decreasing the variance of $\delta\tilde{\boldsymbol{\rho}}'^{\mathrm{u}}$. This is seen with the blue line of Fig. 3c (the variance of $\delta\boldsymbol{b}'^{\mathrm{u}}$ in GB+VR−) being smaller than the green line (GB+VR−), and it explains why the geostrophic response in Fig. 7b is larger than in Fig. 7h. The patterns
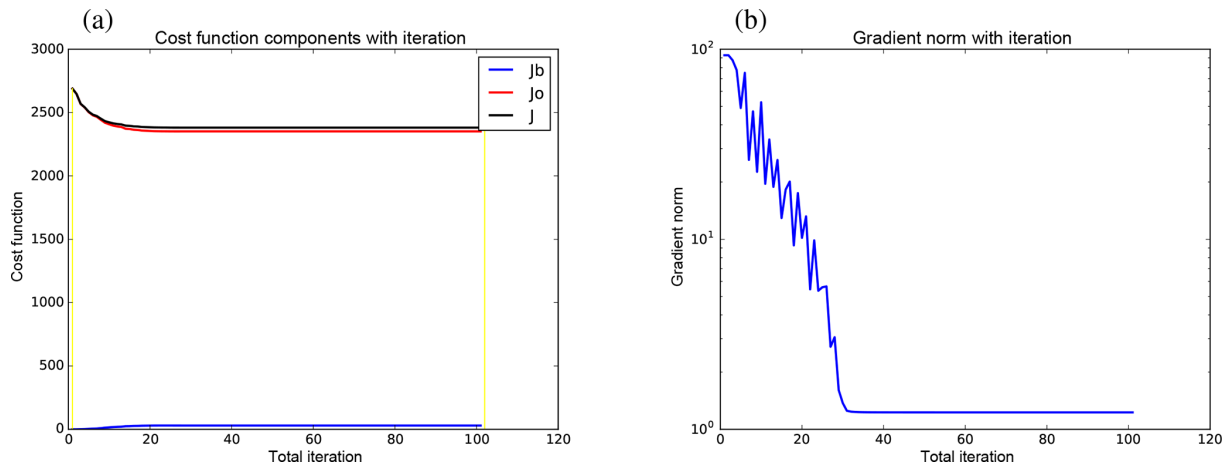
Figure 9. Cost function statistics as a function of inner-loop iteration from the seventh cycle of the GB+VR+ experiment. Panel **(a)** shows the cost function contributions: J^b (blue), J^o (red), and $J = J^b + J^o$ (black). The yellow lines mark the 0th and 100th (the maximum iteration) iterations. Panel **(b)** shows the norm of the cost function gradient.

in the $\delta\tilde{\rho}' - \delta b'$ covariances, Fig. 7c and i, have similar patterns but are larger in panel (c) for similar reasons.

For each experiment, the initial background is found by perturbing the initial truth in a way that is consistent with the **B** matrix of each experiment (see last bullet point in Sect. 4.6). All other aspects of the assimilation are identical in all three experiments: 3DFGAT with a 1 h window is used for 30 DA cycles. The observations are identical in all cases, which are generated from the truth run with an error standard deviation of 0.0015. There are 2520 observations of $\tilde{\rho}'$ in each cycle, which are spread over a 20×18 grid of points broadly covering the domain, and over seven times ($t = 0, 600, 1200, 1800, 2400, 3000, 3600$ s). Only one outer loop of 3DFGAT is executed per cycle with 100 inner-loop iterations. This comparison is intended to be only a brief demonstration of the ABC-DA system, and we leave a more in-depth study to later papers.

5.2 Results for experiment GB+VR+ (control experiment)

Figure 9 shows how the cost function (panel 9a) and the size of its gradient (panel 9b) reduce with inner-loop iteration of the conjugate gradient procedure for one cycle of the GB+VR+ experiment. The total cost function, J (black in panel 9a), decreases along with the observation part, J^o (red in panel 9a), and the gradient (panel 9b) at the expense of the background part of the cost function J^b (blue in panel 9a), as expected. These results appear to show that the cost function has minimised. Figure 10 shows histograms of a selection of differences in the $\tilde{\rho}'$ observation space (for the same cycle as Fig. 9). The top row concerns differences between the observations and $\mathcal{H}(x^b)$ (panel 9a, where $\mathcal{H}(\bullet)$ is the combined ABC model and observation operator for the observation network described at the end of Sect. 5.1), $\mathcal{H}(x^a)$

(panel 9b), and $\mathcal{H}(x^t)$ (panel 9c). The width (standard deviation) of each is shown in the heading of each panel, showing a slight reduction from panels (9a) to (9b) as expected for the DA to have worked. The analysis is closer to the truth than is the background as expected as revealed with the histograms on the bottom row, which are $\mathcal{H}(x^b) - \mathcal{H}(x^t)$ (panel 9d) and $\mathcal{H}(x^a) - \mathcal{H}(x^t)$ (panel 9e).

Diagnostics collected over all 30 cycles for each variable in this experiment are shown in Fig. 11. The left panels show how the root-mean-squared (rms) values of the fields vary in time for three versions of the fields: the truth (solid lines), the analysed fields (dotted), and a free forecast starting from the original background (dashed). For all fields except w, the assimilation overestimates the rms field values, although the assimilation does often follow the oscillations in the true trajectory. The overestimation in the assimilation is actually nearly always worse than for the free background for v, is sometimes worse and sometimes better for u and b', but it is nearly always better for $\tilde{\rho}'$ (recall that only observations of $\tilde{\rho}'$ are assimilated). The right panels show the rms errors (RMSEs) of the analysed (dotted) and free (dashed) fields, which we do know in such twin experiments. In the first few cycles the analyses are improved compared to the free run in all variables, but there comes a time when the assimilation becomes worse than the free run for most fields (especially v and b'). The assimilation stays better than the free run for $\tilde{\rho}'$ over all 30 cycles, and the background error covariances imposed between $\tilde{\rho}'-v$ and $\tilde{\rho}'-b'$ (panels b and c of Fig. 7) do not appear to be of benefit to v and b', which is surprising. Note that there are no assimilation jumps in the u field, because it is neither an observed variable nor correlated to $\tilde{\rho}'$ in the **B** matrix.

At $t = 0$ the two trajectories of the same colour in each right panel represent background (dashed) and analysis (dotted) errors and so can be compared. For all variables (except
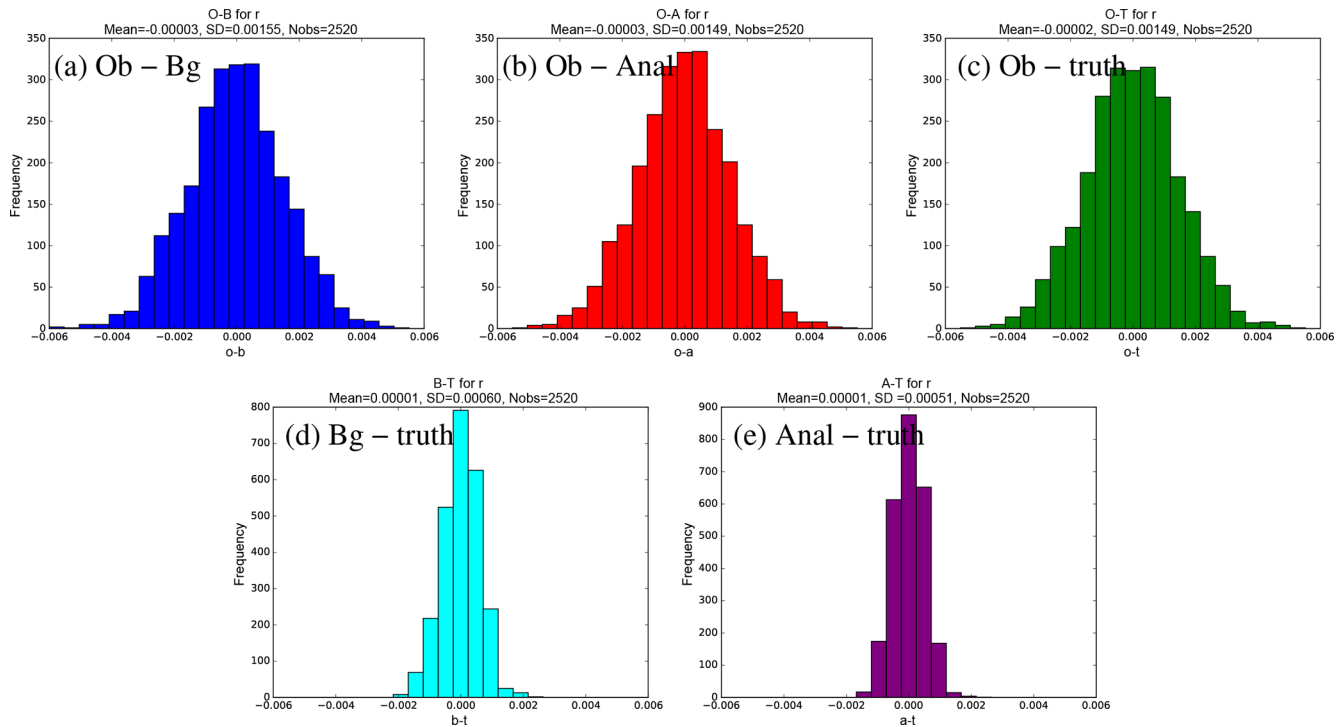
Figure 10. (a–c) Frequency histograms of $y - \mathcal{H}(x)$ for the seventh cycle of the GB+VR+ experiment, where x is **(a)** x^{b}, **(b)** x^{a}, and **(c)** x^{t}. **(d, e)** Histograms of $\mathcal{H}(x) - \mathcal{H}(x^{\mathrm{t}})$, where x is **(d)** x^{b} and **(e)** x^{a}. Observations of $\tilde{\rho}'$ are assimilated.

u and w), these two points are different at $t = 0$, with smaller analysis errors than background errors. This indicates that the DA does add improvement early in the cycling. Even though the analyses are closer to the observations than the backgrounds, the errors in the assimilated variables do grow with time and most of the analysis increments (the jumps at the yellow cycle boundaries) do act to increase the RMSE throughout the experiment. There are a number of possible reasons for this degradation, e.g. a poor **B** matrix formulation for this model (including the use of balance constraints in the CVT), an inappropriate population of forecasts used to calibrate the **B** matrix, the use of 3DFGAT rather than 4DVar, or the particular choice of observation network for this convective-scale system. Given that the assimilation performs well at $t = 0$ (where the error in the initial background is consistent with the **B** matrix formulation by design), it could be that, as the cycling progresses, the nature of actual background errors becomes very different to that assumed by the **B** matrix. Confirming this hypothesis would require recalibrating the **B** matrix, which is outside the scope of the present paper. We focus now on the two remaining experiments described in Sect. 5.1 to see if any changes can be made by adjusting parts of **B**.

5.3 Results for experiment GB−

Diagnostics for the GB− experiment are shown in Fig. 12. Comparing the rms values (left) with the GB+VR+ experi-

ment, the analysis RMS values become further from the truth for u, v, and b'. Note that although the true trajectories are exactly the same as those of GB+VR+, the free trajectories are different as the initial background is consistent with a different **B** matrix. Again comparing to GB+VR+, the RMSE analysis errors (right panels) are generally worse for GB−, but the assimilation is still able to control the $\tilde{\rho}'$ field. This suggests that the **B** matrix for GB− is still not consistent with errors in the 1 h background forecasts over the cycles. Note that there are no assimilation jumps in the u nor v fields, because they are not correlated to $\tilde{\rho}'$ in the **B** matrix.

5.4 Results for experiment GB+VR−

VR is designed to accompany the GB operator, and the configuration that included these two together in the CVT may have contributed to the relatively poor performance in Sect. 5.2. Here we run the experiment that includes the GB step in the CVT but not the VR. Diagnostics for the GB+VR− experiment are shown in Fig. 13. Again, the true state is the same as in the previous two experiments, but the free background forecast is different due to the different **B** matrix from which the initial background state is sampled from. The rms values (left panels) of the assimilation are generally closer to the truth than in GB+VR+ (apart from u where they are about the same). The RMSE values (right panels) are also smaller as a fraction of the error in the free
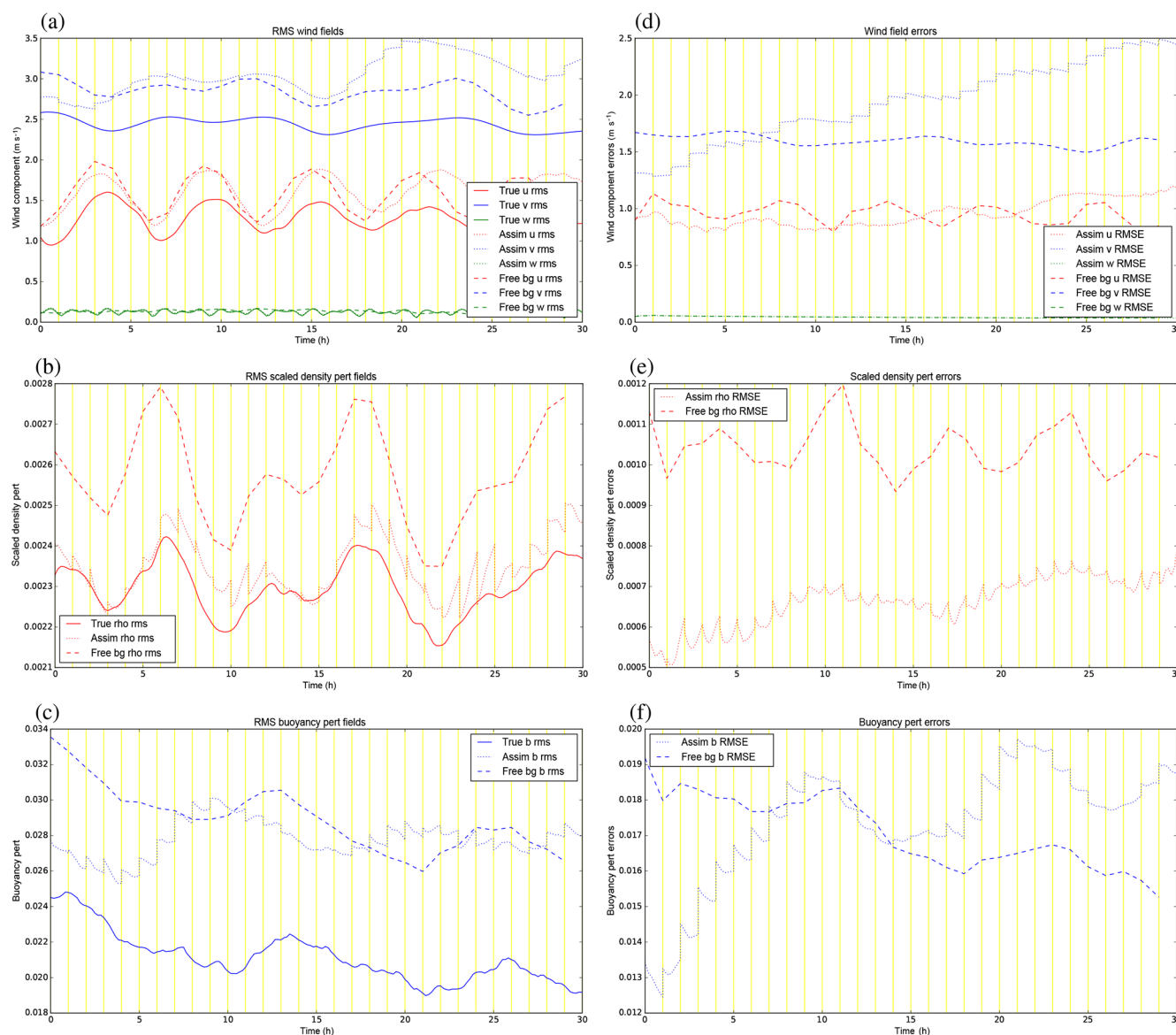
Figure 11. (a–c) The rms fields of **(a)** u, v, w, **(b)** $\tilde{\rho}$', and **(c)** b' for the control experiment, GB+VR+. The solid lines are for the truth trajectory, the dotted lines are for the assimilation trajectory, and the dashed lines are for the free trajectories starting from the initial background. The vertical yellow lines mark the times of the cycle boundaries where the assimilation trajectory is the analysis at the start of each cycle and the background (for the next cycle) at the end. **(d–f)** The rms values of the errors (trajectory minus truth) for the corresponding quantities in **(a–c)**.

run (or at least are smaller for a longer time), although the errors in v do still eventually fail to be controlled by the DA.

The absence of VR does have the effect of weakening the effective strength of the GB between $\delta\tilde{\rho}'$ and δv – compare panels (b) and (h) of Fig. 7. This is because, overall, $\mathbf{R}_\rho$ enhances the balanced scaled density (and so the balanced scaled density is diminished when removing it). It is possible then that it is not the adoption of GB that results in the relatively poor performance of the GB+VR+ assimilation in this system but the use of the VR.

6   Summary

This article is a documentation of ABC-DA, a Var system for use with the convective-scale ABC model (Petrie et al., 2017) to efficiently test new ideas for convective-scale DA. The DA currently has incremental 3DVar and 3DFGAT implemented, although expansion to 4DVar, ensemble-variational, and hybrid methods is possible. The DA system has an expandable observation operator, which is currently configured to assimilate observations of all model variables, and 2D (horizontal) and 3D wind speeds. The choice of observation network is
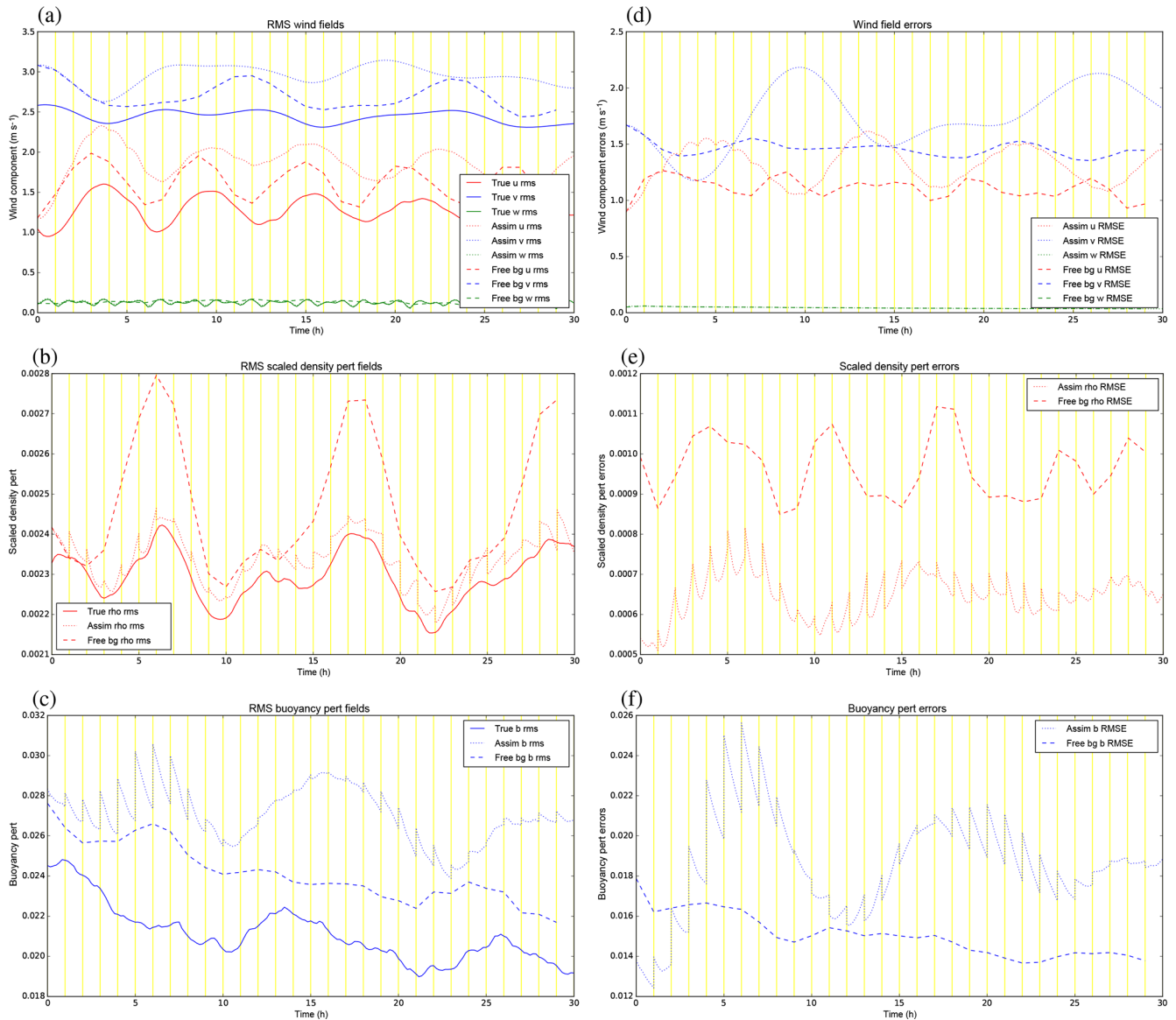
Figure 12. As Fig. 11 but for experiment GB−.

not fixed, and it is possible to specify arbitrary observation networks in space and time. The **B** matrix is modelled in a very compact form by a control variable transform (CVT), which is implemented with a range of options. The current multivariate options include the option to use geostrophic balance (GB) to couple $\delta\tilde{\rho}'$ and δv, hydrostatic balance (HB) to couple $\delta\tilde{\rho}'$ and $\delta b'$, and anelastic balance to couple δu and δw. All of these balances use diagnostic relationships using analytical (rather than statistical) operators. A vertical regression (VR) step is also included (which works with the geostrophic step) to mirror the operational set-up of the Met Office's DA system. While the model increments are δu, δv, δw, $\delta\tilde{\rho}'$, and $\delta b'$, the control parameters are $\delta\psi$, $\delta\chi$, $\delta\tilde{\rho}'^{\mathrm{u}}$, δb^{u}, and δw^{u}, which are assumed to be mutually uncorre-

lated. The transforms that model the spatial covariances for each control parameter are formed of vertical and horizontal parts, which model covariances in these respective directions. There are further options that control how each of these spatial transforms work, including the order in which they are used.

ABC-DA comprises component suites to perform the following tasks:

– run a pure model forecast of any length starting from specified initial conditions,

– investigate linear properties of the equations of motion,

– generate an initial ensemble of states (e.g. for calibration of the **B** matrix),
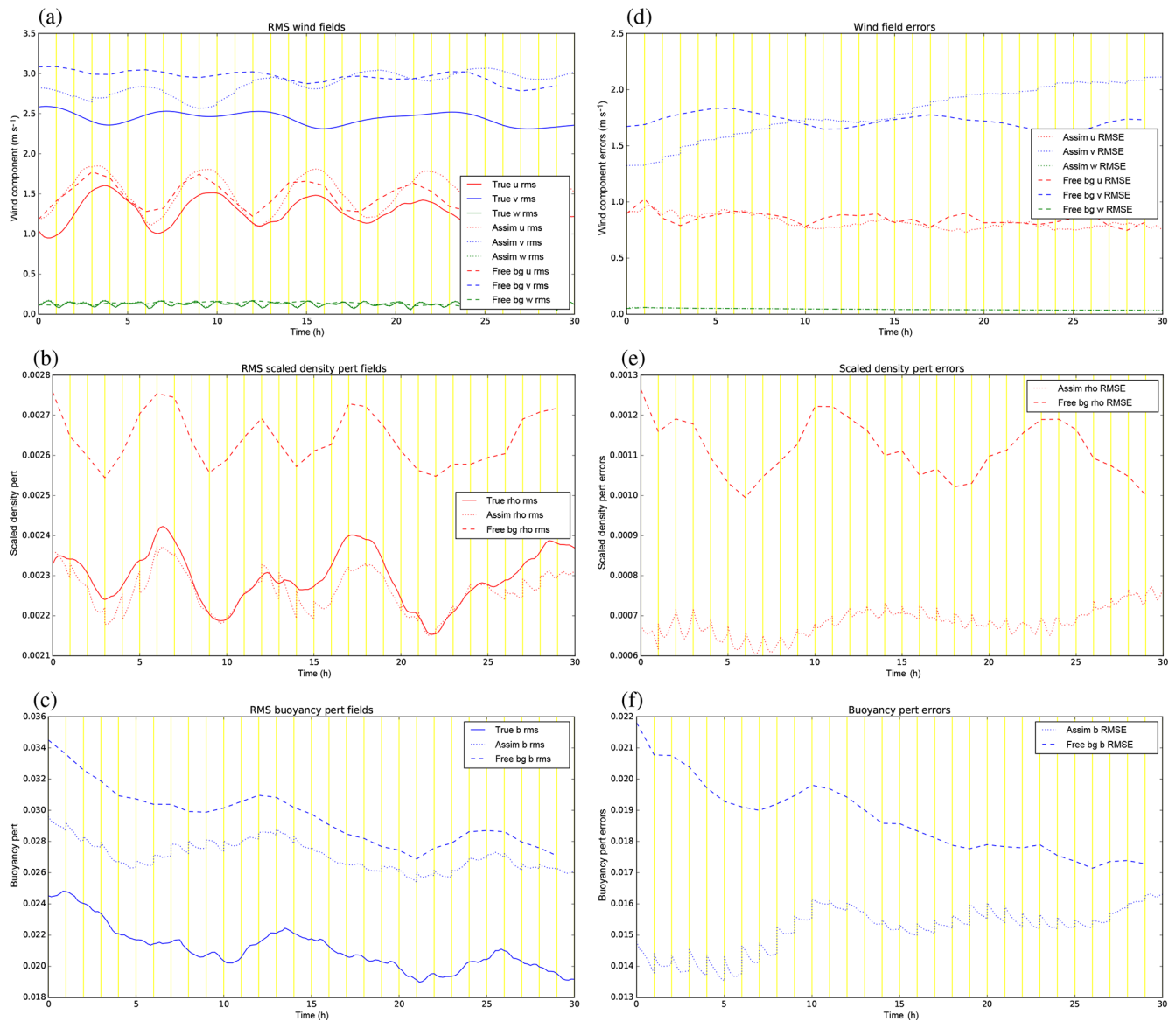
Figure 13. As Fig. 11 but for experiment GB+VR−.

- calibrate the parameters that describe components of the CVT to produce a CVT file (This specifies the **B** matrix for use in the DA program.),

- test the components of the system (e.g. adjoint tests of CVT operators),

- generate "raw" covariance maps using a specified population of states,

- generate implied covariance maps from a specified CVT file,

- generate sample observations (consistent with a specified **R** matrix and truth) and a background state (consistent with a specified **B** matrix and truth),

- perform a cycle of Var with a specified background state, observations, and CVT file,

- perform a run of cycled DA/forecast steps where the analysis of one cycle is used to make a background forecast for the next.

The number-crunching code is written in Fortran 90/95, the scripts are written in Bash, and the example plotting routines are written in Python 2.

A selection of diagnostics of the CVT is provided in this paper, and a brief study is made to illustrate the system's performance when GB (and VR) is switched on and off in the CVT. GB is used to model the background error coupling between $\delta\tilde{\rho}'$ and δv, which is zero when GB is not used, and it has a classical dipole pattern (δv response to a point

perturbation in $\delta\tilde{\rho}'$) when it is used. Exploiting this balance constraint (or not) is investigated as there is a question of whether it has validity in the description of convective-scale background error covariances. VR has the effect of strengthening this geostrophic relationship, because it increases (decreases) the variance of the balanced (unbalanced) contributions to $\delta\tilde{\rho}'$.

Experiments are performed using a set of three 3DFGAT twin experiments by assimilating observations over a 30 h forecast taken as the common truth. The three experiments are (i) using GB and VR (GB+VR+), (ii) not using GB (GB−), and (iii) using GB but not VR (GB+VR−). Each experiment has a different CVT (i.e. a different $\mathbf{U}$) and hence a different implied background error covariance matrix; $\mathbf{B}^{\mathrm{ic}} = \mathbf{U}\mathbf{U}^{\mathrm{T}}$. An initial background state is constructed for each experiment by perturbing the truth at $t = 0$ with $\mathbf{U}\delta\boldsymbol{\chi}$ where $\delta\boldsymbol{\chi}$ is a sample from $N(0, \mathbf{I})$ (equivalent in model space as choosing a sample from $N(0, \mathbf{B}^{\mathrm{ic}})$). The DA/forecast is cycled hourly over 30 cycles with 2520 observations of $\tilde{\rho}'$ per cycle. The results are summarised as follows.

- The GB+VR+ experiment does not result in a systematic reduction of errors in the model variables. In fact, for some variables (namely v and b'), it leads to an increase in error of the analysis above the error level of the free background.

- The GB− experiment represents a degradation in performance of the unobserved variables in that the errors are above the error level of the free background for more time than in GB+VR+.

- The GB+VR− experiment represents the configuration that is the best of those tested. The errors in all variables except v are kept below the error level of the free background.

It is surprising that errors in one of the variables in particular (v) cannot be properly corrected in these experiments. It is postulated that this outcome is not due to an inappropriate $\mathbf{B}$ matrix (for instance in Figs. 11 and 13) but due to the use of 3DFGAT. Separate tests with 3DVar and with all observations made at the analysis times, rather than spread throughout the analysis window (not shown), suggest that observations of $\tilde{\rho}'$ can reduce the error in v. We have learned from these experiments that the nature of the $\mathbf{B}$ matrix can have a significant impact on the performance of the DA. Application of GB in such a convective-scale system has a complicated effect: while experiment GB+VR− is better than experiment GB− overall, the application of GB does systematically increase errors in the v field over the 30 cycles, and "too much" GB (as in experiment GB+VR+) enhances this effect. Similar studies may be done by turning on and off hydrostatic and anelastic balances.

These results highlight a difficulty of the convective-scale DA problem, and they raise more questions about how this system could be improved and how it could contribute to advances in DA. A possible way forward might be to keep the current formulation of the CVT but to calibrate it with data from a different source, either as an ensemble, as differences between pairs of forecasts valid at the same time (the popular National Meteorological Center method; Parrish and Derber, 1992; Berre et al., 2006), or from a single long forecast (the so-called "Canadian quick" method; Polavarapu et al., 2005). Other ways forward might be to replace the traditional geophysical balances with a new balance relationship that may be more suitable at convective scales, such as the diagnostic equation for non-hydrostatic pressure outlined in Sect. 4.3 of Pielke (2002). The CVT may also be adapted so that the DA system acts as an ensemble-variational (EnVar) system, where the analysis increments are expressed as linear combinations of ensemble perturbations (e.g. Lorenc, 2003; Buehner, 2005), as a hybrid-EnVar system (a hybrid of the existing $\mathbf{B}$ matrix and the EnVar; e.g. Wang et al., 2007), or as part of a hybrid gain formulation (e.g. Penny, 2014). These approaches imply flow-dependent background error covariances (e.g. Bannister, 2017), but each would require a separate ensemble DA system parallel to the variational one and would also require localisation techniques to mitigate for rank deficiency of the ensemble. These are entirely possible developments given appropriate resources. Studying the $\mathbf{B}$ matrix is the main application that the author has in mind, although the system could equally well be used to study impacts of new high-resolution observations.

Appendix A: Number of pieces of information to make up a covariance model

This appendix shows the number of pieces of information that specifies the covariance model used in this paper. Since only analytic balance relationships are used (rather than statistically derived ones, which would themselves require calibration from data), components only of the spatial transform ($\mathbf{U}_s$) and the vertical regression ($\mathbf{R}_\rho$) require calibration. Referring to Eq. (31) or Eq. (32), this includes determination of the standard deviation matrix ($\mathbf{\Sigma}$) and components of the vertical ($\mathbf{U}_v = \mathbf{F}_v \mathbf{\Lambda}_v^{1/2}$) and horizontal ($\mathbf{U}_h = \mathbf{F}_h \mathbf{\Lambda}_h^{1/2}$) transforms. The vertical regression, $\mathbf{R}_\rho = \mathbf{C}^{\delta\tilde{\rho}'\delta\tilde{\rho}'^b}\left(\mathbf{C}^{\delta\tilde{\rho}'^b\delta\tilde{\rho}'^b}\right)^{-1}$, requires determination of a non-symmetric and symmetric correlation matrix. Table A1 shows this information, concluding that 78 030 pieces of information are needed. In this paper, a super-ensemble of 260 members is used to estimate this background error covariance model. This provides $260 \times 5 \times 360 \times 60 = 28\,080\,000$ pieces of information, vastly more than the number of elements needed for the covariance model. Compare this to $n(1+n)/2$ elements to determine a full background error covariance matrix, where $n = 5 \times 360 \times 60$ is the number of elements in the model state vector. This evaluates to $5\,832\,054\,000$, requiring 54 000 ensemble members to estimate these elements at the very least. While this is possible, it is not practical at present.

Appendix B: Vertical regression of balanced scaled density

The direct calculation of the geostrophically balanced scaled density from Eq. (2a) (in practice found via the streamfunction as in Eq. 15) leads to difficulties with regards to the vertical continuity of the resulting $\delta\tilde{\rho}'^b(x, z, t)$ (Lorenc et al., 2000). If a function (arbitrarily chosen but independent of x, i.e. $p(z, t)$) is added to $\delta\tilde{\rho}'^b$, then $\delta\tilde{\rho}'^b + p$ is also a solution of Eq. (2a). This will potentially allow damaging jumps to appear with short vertical scale. According to geostrophic adjustment theory, streamfunction (and hence the scaled density computed from it) is not balanced for short vertical scales (e.g. Cullen, 2003) and so the output from Eq. (15) is smoothed in the vertical. This is done by modifying the balanced scaled density with a regression operator $\mathbf{R}_\rho$.

From here on in this appendix, we drop the ρ subscript on $\mathbf{R}_\rho$ for brevity as the subscript will be used to indicate matrix element. We also, for brevity, drop the tilde, prime, and δ notations on the scaled density fields, remembering that the fields are implicitly perturbations. One strategy for $\mathbf{R}$ is to introduce similar vertical length scales as those in the forecasts of total scaled density, $\delta\rho$. We therefore design $\mathbf{R}$ to minimise the following penalty:

$$\mathbf{R} = \underset{\mathbf{R}}{\mathrm{argmin}}\,\mathcal{J}[\mathbf{R}]$$
$$= \frac{1}{2}\frac{1}{N}\sum_{i=1}^{N}\left(\mathbf{R}\boldsymbol{\rho}^b(i) - \boldsymbol{\rho}(i)\right)^{\mathrm{T}}\left(\mathbf{R}\boldsymbol{\rho}^b(i) - \boldsymbol{\rho}(i)\right), \qquad \text{(B1)}$$

where $\boldsymbol{\rho}^b(i)$ and $\boldsymbol{\rho}(i)$ are the ith members of an N-member population of balanced and total scaled density perturbations, respectively. To find the $\mathbf{R}$ that minimises $\mathcal{J}$, we solve the equation $\partial\mathcal{J}/\partial\mathbf{R} = 0$, which is a matrix equation, with element a, b being $\partial\mathcal{J}/\partial\mathbf{R}_{ab} = 0$.

Table A1. The numbers of components of the spatial transforms that need to be estimated from data. The numbers of elements in the third column are for the standard ABC system of 360 longitudes and 60 levels. Note that for $\mathbf{\Lambda}_h^{1/2}$, 180 wavenumbers are specified rather than 360 as the negative wavenumbers are assumed to have the same variances as the positive ones.

Matrix	Description	Elements to be determined
$\mathbf{\Sigma}$	Standard deviations (level-dependent only)	60
$\mathbf{F}_v$	Eigenvectors of vertical transforms	60×60
$\mathbf{\Lambda}_v^{1/2}$	Eigenvalues of vertical transforms	60
$\mathbf{F}_h$	Horizontal Fourier transform	0
$\mathbf{\Lambda}_h^{1/2}$	Horizontal variances of each parameter vertical mode	180×60
	Total per control parameter (sum of above)	14 520
	For all five control parameters	72 600
$\mathbf{C}^{\delta\tilde{\rho}'\delta\tilde{\rho}'^b}$	Non-symmetric term in $\mathbf{R}_\rho$	60×60
$\mathbf{C}^{\delta\tilde{\rho}'^b\delta\tilde{\rho}'^b}$	Symmetric term in $\mathbf{R}_\rho$	$60 \times 61/2$
	Grand total	78 030

It is convenient to expand the elements of Eq. (B1):

$$\mathcal{J}[\mathbf{R}] = \frac{1}{2}\frac{1}{N}\sum_{i=1}^{N}\sum_{\alpha}\left(\sum_{\beta}\mathbf{R}_{\alpha\beta}\rho_{\beta}^{\mathrm{b}}(i) - \rho_{\alpha}(i)\right)^{2}, \qquad (B2)$$

where all subscripts indicate matrix or vector element. Differentiating with respect to $\mathbf{R}_{ab}$ gives

$$\begin{aligned}
\frac{\partial\mathcal{J}}{\mathbf{R}_{ab}} &= \frac{1}{N}\sum_{i=1}^{N}\sum_{\alpha}\left(\sum_{\beta}\mathbf{R}_{\alpha\beta}\rho_{\beta}^{\mathrm{b}}(i) - \rho_{\alpha}(i)\right)\\
&\quad \frac{\partial}{\mathbf{R}_{ab}}\left(\sum_{\beta'}\mathbf{R}_{\alpha\beta'}\rho_{\beta'}^{\mathrm{b}}(i)\right)\\
&= \frac{1}{N}\sum_{i=1}^{N}\sum_{\alpha}\left(\sum_{\beta}\mathbf{R}_{\alpha\beta}\rho_{\beta}^{\mathrm{b}}(i) - \rho_{\alpha}(i)\right)\sum_{\beta'}\delta_{a\alpha}\delta_{b\beta'}\rho_{\beta'}^{\mathrm{b}}(i)\\
&= \frac{1}{N}\sum_{i=1}^{N}\left(\sum_{\beta}\mathbf{R}_{a\beta}\rho_{\beta}^{\mathrm{b}}(i) - \rho_{a}(i)\right)\rho_{b}^{\mathrm{b}}(i),
\end{aligned}$$

$$(B3)$$

where $\delta_{\alpha\beta}$ in the second line is the Kronecker delta function. Now this result is put back into matrix notation. Note that $\left(\sum_{\beta}\mathbf{R}_{a\beta}\rho_{\beta}^{\mathrm{b}}(i) - \rho_{a}(i)\right)\rho_{b}^{\mathrm{b}}(i)$ is the a,bth element of the outer product matrix $\left(\mathbf{R}\rho^{\mathrm{b}}(i) - \rho(i)\right)\left(\rho^{\mathrm{b}}(i)\right)^{\mathrm{T}}$, so Eq. (B3) is

$$\frac{\partial\mathcal{J}}{\mathbf{R}} = \frac{1}{N}\sum_{i=1}^{N}\left(\mathbf{R}\rho^{\mathrm{b}}(i) - \rho(i)\right)\left(\rho^{\mathrm{b}}(i)\right)^{\mathrm{T}}.$$

Setting this to zero to solve for the optimal regression matrix gives

$$\mathbf{R}\frac{1}{N}\sum_{i=1}^{N}\left(\rho^{\mathrm{b}}(i)\right)\left(\rho^{\mathrm{b}}(i)\right)^{\mathrm{T}} = \frac{1}{N}\sum_{i=1}^{N}\left(\rho^{\mathrm{b}}\rho(i)\right)\left(\rho^{\mathrm{b}}(i)\right)^{\mathrm{T}}.$$

The outer product matrix – summed and divided by N – on the left-hand side is identified as the covariance matrix of the balanced scaled density perturbations with themselves ($\mathbf{C}^{\rho^{\mathrm{b}}\rho^{\mathrm{b}}}$), and the corresponding terms on the right-hand side is the covariance matrix between the total scaled density perturbations and the balanced scaled density perturbations ($\mathbf{C}^{\rho\rho^{\mathrm{b}}}$). The optimal regression matrix is then

$$\mathbf{R} = \mathbf{C}^{\rho\rho^{\mathrm{b}}}\left(\mathbf{C}^{\rho^{\mathrm{b}}\rho^{\mathrm{b}}}\right)^{-1}. \qquad (B4)$$

Appendix C: Estimation of characteristic length scales

There is a well-defined length scale associated with a plane wave of the form $\exp i\kappa x$ (κ is wavenumber, x is position), which is the wavelength $l = 2\pi/\kappa$. For a periodic domain of length L, the allowed wavenumbers are $\kappa = 2k\pi/L$ (k wavenumber index, integer), and so the wavelength is L/k. There is no similar analytical relationship between the vertical mode index, ν, and the vertical length scale, l_{v}, but it is possible to estimate the characteristic length scale as follows. Suppose that $f(z)$ is an oscillatory function (e.g. a particular vertical eigenvector (column of $\mathbf{F}_{\mathrm{v}}$) of one of the vertical covariance matrices introduced in Sect. 4.2.3). Let its Fourier transform be $\overline{f}(m)$, where m is the spectral co-ordinate (i.e. $f(z) \propto \sum_{m}\overline{f}(m)\exp i2m\pi z/L_{\mathrm{v}}$, where L_{v} is the vertical extent of the model). Treating $\left|\overline{f}(m)\right|^{2}$ as the weight of the contribution from wavenumber index m, the characteristic value of m is

$$\langle m \rangle = \sum_{m}\left|\overline{f}(m)\right|^{2}m \Big/ \sum_{m}\left|\overline{f}(m)\right|^{2}. \qquad (C1)$$

This translates to a vertical length scale of $l_{\mathrm{v}} = L_{\mathrm{v}}/\langle m \rangle$. This is the procedure used to make Fig. 6b. We note that the Fourier transform assumes periodicity, which is not a property of the fields in the vertical direction, but since we are after only a rough estimate of l_{v}, we assume that this inconsistency does not have a major impact.

References

Baldauf, M., Seifert, A., Förstner, J., Majewski, D., Raschendorfer, M., and Reinhardt, T.: Operational convective-scale numerical weather prediction with the COSMO model: Description and sensitivities, Mon. Weather Rev., 139, 3887–3905, 2011.

Bannister, R.: A review of operational methods of variational and ensemble-variational data assimilation, Q. J. Roy. Meteor. Soc., 143, 607–633, https://doi.org/10.1002/qj.2982, 2017.

Bannister, R. N.: A review of forecast error covariance statistics in atmospheric variational data assimilation. I: Characteristics and measurements of forecast error covariances, Q. J. Roy. Meteor. Soc., 134, 1951–1970, 2008a.

Bannister, R. N.: A review of forecast error covariance statistics in atmospheric variational data assimilation. II: Modelling the forecast error covariance statistics, Q. J. Roy. Meteor. Soc., 134, 1971–1996, 2008b.

Bannister, R. N.: ABC-DA model software, GitHub available at: https://github.com/rossbannister/ABC-DA_1.4da (last access: 24 August 2020), 2019.

Bannister, R. N.: Sample of initial condition data for the ABC simplified atmospheric model and data assimilation system (vn1.4da) (Version vn1.4da), Zenodo, https://doi.org/10.5281/zenodo.3946359, 2020.

Bannister, R. N., Migliorini, S., and Dixon, M.: Ensemble prediction for nowcasting with a convection-permitting model–II: Forecast error statistics, Tellus A, 63, 497–512, 2011.

Bannister, R. N., Chipilski, H., and Martinez-Alvarado, O.: Techniques and challenges in the assimilation of atmospheric water observations for numerical weather prediction towards convective scales, Q. J. Roy. Meteor. Soc., 146, 1–48, https://doi.org/10.1002/qj.3652, 2020.

Bartello, P. and Mitchell, H. L.: A continuous three-dimensional model of short-range forecast error covariances, Tellus A, 44, 217–235, 1992.

Berre, L.: Estimation of synoptic and mesoscale forecast error covariances in a limited-area model, Mon. Weather Rev., 128, 644–667, 2000.

Berre, L., Ecaterina Ştefănescu, S., and Belo Pereira, M.: The representation of the analysis effect in three error simulation techniques, Tellus A, 58, 196–209, 2006.

Brousseau, P., Seity, Y., Ricard, D., and Léger, J.: Improvement of the forecast of convective activity from the AROME-France system, Q. J. Roy. Meteor. Soc., 142, 2231–2243, 2016.

Buehner, M.: Ensemble-derived stationary and flow-dependent background-error covariances: Evaluation in a quasi-operational NWP setting, Q. J. Roy. Meteor. Soc., 131, 1013–1043, 2005.

Caron, J.-F. and Fillion, L.: An examination of background error correlations between mass and rotational wind over precipitation regions, Mon. WeatherRev., 138, 563–578, 2010.

Chen, Y., Rizvi, S. R., Huang, X.-Y., Min, J., and Zhang, X.: Balance characteristics of multivariate background error covariances and their impact on analyses and forecasts in tropical and Arctic regions, Meteorol. Atmos. Phys., 121, 79–98, 2013.

Chen, Y., Xia, X., Min, J., Huang, X.-Y., and Rizvi, S. R.: Balance characteristics of multivariate background error covariance for rainy and dry seasons and their impact on precipitation forecasts of two rainfall events, Meteorol. Atmos. Phys., 128, 579–600, 2016.

Clark, P., Roberts, N., Lean, H., Ballard, S. P., and Charlton-Perez, C.: Convection-permitting models: a step-change in rainfall forecasting, Meteorol. Appl., 23, 165–181, 2016.

Courtier, P., Thépaut, J.-N., and Hollingsworth, A.: A strategy for operational implementation of 4D-Var, using an incremental approach, Q. J. Roy. Meteor. Soc., 120, 1367–1387, 1994.

Cullen, M.: Four-dimensional variational data assimilation: A new formulation of the background-error covariance matrix based on a potential-vorticity representation, Q. J. Roy. Meteor. Soc., 129, 2777–2796, 2003.

Cullen, M. and Davies, T.: A conservative split-explicit integration scheme with fourth-order horizontal advection, Q. J. Roy. Meteor. Soc., 117, 993–1002, 1991.

Cullen, M. J. P.: A mathematical theory of large-scale atmosphere/ocean flow, Imperial College Press, London, UK, 2006.

Deckmyn, A. and Berre, L.: A wavelet approach to representing background error covariances in a limited-area model, Mon. Weather Rev., 133, 1279–1294, 2005.

Derber, J. and Bouttier, F.: A reformulation of the background error covariance in the ECMWF global data assimilation system, Tellus A, 51, 195–221, 1999.

Dimet, F.-X. L. and Talagrand, O.: Variational algorithms for analysis and assimilation of meteorological observations: theoretical aspects, Tellus A, 38, 97–110, 1986.

Fabry, F. and Sun, J.: For how long should what data be assimilated for the mesoscale forecasting of convection and why? Part I: On the propagation of initial condition errors and their implications for data assimilation, Mon. Weather Rev., 138, 242–255, 2010.

Gauthier, P., Buehner, M., and Fillion, L.: Background-error statistics modelling in a 3D variational data assimilation scheme: Estimation and impact on the analyses, in: Proc. ECMWF Workshop on Diagnosis of Data Assimilation Systems, ECMWF (the European Centre for Medium Range Weather Forecasts), 131–145, 1999.

Golub, G. H. and Van Loan, C. F.: Matrix computations. 1996, Johns Hopkins University, Press, Baltimore, MD, USA, 374–426, 1996.

Gustafsson, N., Berre, L., Hörnquist, S., Huang, X.-Y., Lindskog, M., Navascues, B., Mogensen, K., and Thorsteinsson, S.: Three-dimensional variational data assimilation for a limited area model, Tellus A, 53, 425–446, 2001.

Houtekamer, P. and Zhang, F.: Review of the Ensemble Kalman Filter for Atmospheric Data Assimilation, Mon. Weather Rev., 144, 4489–4532, 2016.

Ingleby, N. B.: The statistical structure of forecast errors and its representation in the Met. Office global 3-D variational data assimilation scheme, Q. J. Roy. Meteor. Soc., 127, 209–231, 2001.

Kalnay, E.: Atmospheric modeling, data assimilation and predictability, Cambridge University Press, Cambridge, UK, 2002.

Katz, D., Lawless, A. S., Nichols, N. K., Cullen, M., and Bannister, R. N.: Correlations of control variables in variational data assimilation, Q. J. Roy. Meteor. Soc., 137, 620–630, 2011.

Lawless, A.: A note on the analysis error associated with 3D-FGAT, Q. J. Roy. Meteor. Soc., 136, 1094–1098, 2010.

Lee, M.-S., Barker, D., Huang, W., and Kuo, Y.-H.: First guess at appropriate time (FGAT) with WRF 3DVAR, in: WRF/MM5 Users Workshop, 22–25, available at: https://www.researchgate.net/publication/242428553_First_Guess_at_Appropriate_Time_FGAT_with_WRF_3DVAR (last accees: 24 August 2020), 2004.

Lewis, J. M., Lakshmivarahan, S., and Dhall, S.: Dynamic data assimilation: a least squares approach, vol. 13, Cambridge University Press, 2006.

Liu, C., Xiao, Q., and Wang, B.: An ensemble-based four-dimensional variational data assimilation scheme. Part I: Technical formulation and preliminary test, Mon. Weather Rev., 136, 3363–3373, 2008.

Lorenc, A., Ballard, S., Bell, R., Ingleby, N., Andrews, P., Barker, D., Bray, J., Clayton, A., Dalby, T., Li, D., Payne, T., and Saunders, F.: The Met. Office global three-dimensional variational data assimilation scheme, Q. J. Roy. Meteor. Soc., 126, 2991–3012, 2000.

Lorenc, A. C.: The potential of the ensemble Kalman filter for NWP: comparison with 4D-Var, Q. J. Roy. Meteor. Soc., 129, 3183–3203, 2003.

Parrish, D. F. and Derber, J. C.: The National Meteorological Center's spectral statistical-interpolation analysis system, Mon. Weather Rev., 120, 1747–1763, 1992.

Penny, S. G.: The hybrid local ensemble transform Kalman filter, Mon. Weather Rev., 142, 2139–2149, 2014.

Pereira, M. B. and Berre, L.: The use of an ensemble approach to study the background error covariances in a global NWP model, Mon. Weather Rev., 134, 2466–2489, 2006.

Petrie, R. E., Bannister, R. N., and Cullen, M. J. P.: The "ABC model": a non-hydrostatic toy model for use in convective-scale data assimilation investigations, Geosci. Model Dev., 10, 4419–4441, https://doi.org/10.5194/gmd-10-4419-2017, 2017.

Pielke, R.: Mesoscale Meteorological Modeling, Academic Press, San Diego, CA, USA, 2002.

Polavarapu, S., Ren, S., Rochon, Y., Sankey, D., Ek, N., Koshyk, J., and Tarasick, D.: Data assimilation with the Canadian middle atmosphere model, Atmos.-Ocean, 43, 77–100, 2005.

Schwartz, C. S. and Liu, Z.: Convection-permitting forecasts initialized with continuously cycling limited-area 3DVar, ensemble Kalman filter, and "hybrid" variational–ensemble data assimilation systems, Mon. Weather Rev., 142, 716–738, 2014.

Sun, J., Xue, M., Wilson, J. W., Zawadzki, I., Ballard, S. P., Onvlee-Hooimeyer, J., Joe, P., Barker, D. M., Li, P.-W., Golding, B., Xu, M., and Pinto, J.: Use of NWP for nowcasting convective precipitation: Recent progress and challenges, B. Am. Meteorol. Soc., 95, 409–426, 2014.

Sun, J., Wang, H., Tong, W., Zhang, Y., Lin, C.-Y., and Xu, D.: Comparison of the impacts of momentum control variables on high-resolution variational data assimilation and precipitation forecasting, Mon. Weather Rev., 144, 149–169, 2016.

Tang, Y., Lean, H. W., and Bornemann, J.: The benefits of the Met Office variable resolution NWP model for forecasting convection, Meteorol. Appl., 20, 417–426, 2013.

Wang, X., Snyder, C., and Hamill, T. M.: On the theoretical equivalence of differently proposed ensemble-3D-Var hybrid analysis schemes, Mon. Weather Rev., 135, 222–227, 2007.

Wlasak, M. and Cullen, M.: Modelling static 3-D spatial background error covariances–the effect of vertical and horizontal transform order, Adv. Sci. Res., 11, 63–67, 2014.

Yano, J.-I., Ziemiański, M. Z., Cullen, M., Termonia, P., Onvlee, J., Bengtsson, L., Carrassi, A., Davy, R., Deluca, A., Gray, S. L., Homar, V., Köhler, M., Krichak, S., Michaelides, S., Phillips, V. T. J., Soares, P. M. M., and Wyszogrodzki, A. A.: Scientific challenges of convective-scale numerical weather prediction, B. Am. Meteorol. Soc., 99, 699–710, 2018.