Building tangent-linear and adjoint models for data assimilation with neural networks

Samuel Edward Hatfield\textsuperscript{1}, Matthew Chantry\textsuperscript{2}, Peter Dominik Dueben\textsuperscript{1}, Philippe Lopez\textsuperscript{1}, Alan Jon Geer\textsuperscript{3}, and Tim N Palmer\textsuperscript{4}

\textsuperscript{1}ECMWF
\textsuperscript{2}Atmospheric, Oceanic and Planetary Physics, University of Oxford
\textsuperscript{3}European Centre for Medium Range Weather Forecasts
\textsuperscript{4}Oxford University

November 24, 2022

Abstract

We assess the ability of neural network emulators of physical parametrization schemes in numerical weather prediction models to aid in the construction of linearised models required by 4D-Var data assimilation. Neural networks can be differentiated trivially, and so if a physical parametrization scheme can be accurately emulated by a neural network then its tangent-linear and adjoint versions can be obtained with minimal effort, compared with the standard paradigms of manual or automatic differentiation of the model code. Here we demonstrate this idea by emulating the non-orographic gravity wave drag parametrization scheme in an atmospheric model with a neural network, and deriving its tangent-linear and adjoint models. We demonstrate that these neural network-derived tangent-linear and adjoint models not only pass the standard consistency tests but also can be used successfully to do 4D-Var data assimilation. This technique holds the promise of significantly easing maintenance of tangent-linear and adjoint codes in weather forecasting centres, if accurate neural network emulators can be constructed.
Building tangent-linear and adjoint models for data assimilation with neural networks

Sam Hatfield¹, Matthew Chantry², Peter Dueben¹, Philippe Lopez¹, Alan Geer¹, Tim Palmer²

¹European Centre for Medium-Range Weather Forecasts, Shinfield Park, Reading
²Atmospheric, Oceanic and Planetary Physics, University of Oxford, Oxford, United Kingdom

Key Points:

• Neural network emulators of physical parametrization schemes can be used to easily construct tangent-linear and adjoint models.
• These tangent-linear and adjoint models can be used to perform data assimilation in state-of-the-art weather forecasting models.
• Neural network emulation may allow for a better representation of linearised physical processes in 4D-Var data assimilation.

Corresponding author: Sam Hatfield, samuel.hatfield@ecmwf.int
Abstract

We assess the ability of neural network emulators of physical parametrization schemes in numerical weather prediction models to aid in the construction of linearised models required by 4D-Var data assimilation. Neural networks can be differentiated trivially, and so if a physical parametrization scheme can be accurately emulated by a neural network then its tangent-linear and adjoint versions can be obtained with minimal effort, compared with the standard paradigms of manual or automatic differentiation of the model code. Here we demonstrate this idea by emulating the non-orographic gravity wave drag parametrization scheme in an atmospheric model with a neural network, and deriving its tangent-linear and adjoint models. We demonstrate that these neural network-derived tangent-linear and adjoint models not only pass the standard consistency tests but also can be used successfully to do 4D-Var data assimilation. This technique holds the promise of significantly easing maintenance of tangent-linear and adjoint codes in weather forecasting centres, if accurate neural network emulators can be constructed.

Plain Language Summary

The neural network is an algorithm developed in the field of artificial intelligence that can in principle learn the relationship between any two variables, provided you give it enough real-world data. There are countless applications for such an algorithm in the field of weather and climate simulation. The application that we focus on here is to use the neural network as a replacement for one part of a weather simulation. Essentially, you train the neural network so that it can replicate exactly the part that it replaces. Then, when you want to run the simulation, you use the neural network instead because it’s much faster. Other studies have already demonstrated that this technique can be applied in weather simulations. What we show here, however, is that neural networks can also be used to calculate the slope of the line relating two variables. This makes them especially useful in helping to construct the initial conditions for weather forecasts, through a process known as data assimilation.

1 Introduction

The past few years have seen a surge in interest in machine learning in numerous scientific fields, and numerical weather prediction (NWP) is no exception. The field of machine learning provides a variety of inference and categorisation tools which can be
applied to a diverse set of problems, but perhaps the most relevant from a NWP perspective is regression. The regression task in its most generic form — that is, to learn the relation between a set of input and output vector pairs — has numerous potential applications throughout a NWP pipeline, from preprocessing of observations to postprocessing of forecasts. In this paper, we test whether the emulation of physical parametrization schemes in weather and climate models by one of the most promising regression techniques, the neural network, can also help us to perform data assimilation, in which observations are used to construct the initial conditions for a forecast through blending with a numerical model.

The mainstay of data assimilation at the European Centre for Medium-Range Weather Forecasts (ECMWF) and a number of other operational NWP centres has been the four-dimensional variational technique (4D-Var) for over two decades (Rabier, 1998). With 4D-Var, one constructs a cost function that measures the least-squares fit of a model trajectory to many observations distributed across a window of time (an “assimilation window”). Beginning from a prior guess of the control variable, the model state at the beginning of the window, one then proceeds to minimise this cost function through a gradient-descent algorithm. Given that the control variable is the model state at the beginning of the assimilation window, 4D-Var requires a way to propagate gradients with respect to later states of the model trajectory backwards in time, a role fulfilled by the so-called adjoint model. Typical formulations of 4D-Var adopt the “incremental approach”, in which the control variable is not a model state but in fact a model state perturbation with respect to the prior guess model state (Courtier et al., 1994). To advance a model state perturbation forward in time we require the so-called tangent-linear model, which is closely related to the adjoint model as explained below.

The problems of regression by deep learning and 4D-Var data assimilation bear a resemblance, and in some cases are formally the same (Geer, 2021). In both cases one considers a forward model depending on some parameters which maps some inputs (“features”) to some outputs (“labels”). Additionally, both problems are formulated in terms of the minimisation of a “cost function”. For data assimilation, the control variable for this cost function is the model state, whereas for deep learning, the control variable is the vector of model parameters. The minimisation of this cost function for deep learning problems is typically accomplished through backpropagation, a gradient-descent approach very similar to that used by 4D-Var. Given the similarities of these two fields,
the long history of successful 4D-Var data assimilation within NWP and the extensive interest in novel deep learning techniques, an exploration of synergies between the two techniques would likely be fruitful.

The success of the 4D-Var approach is crucially dependent on the construction of accurate tangent-linear and adjoint versions of every model component, especially for the physical parametrization schemes that represent, for example, the unresolved part of the fluid dynamics. Such an effort is a focus of significant activity at centres that use 4D-Var, such as ECMWF and the UK Met Office (Payne, 2021). Constructing these linearised models can either be achieved by manually differentiating the nonlinear code or by using an automatic differentiation tool. The comprehensive linear physics package of ECMWF was constructed through the former, manual approach after first constructing simplified and “regularised” versions of the nonlinear physics, which are computationally cheaper and give more stable behaviour after linearisation (Janisková & Lopez, 2013). This technique has been used successfully so far for the radiation, vertical diffusion, unresolved gravity wave drag, convection and clouds and precipitation schemes, but could present limits to the overall scalability of the data assimilation system in the future (Bauer et al., 2020). Due to the strict nature of the linear code, namely that the nonlinear, tangent-linear and adjoint models must be formulated in a mutually consistent way, it is harder to port these model components to novel computational hardware, such as graphics processing units, in their current form. The main alternative to 4D-Var, the ensemble Kalman filter, does not require tangent-linear or adjoint models (Hamrud et al., 2015; Bonavita et al., 2015). However, this technique suffers from its own scalability problems so it is no panacea. It is therefore worthwhile to explore means to keep the 4D-Var algorithm competitive by alleviating the aforementioned scalability issues.

In this paper we propose a novel method of constructing tangent-linear and adjoint models for use in 4D-Var that relies on machine learning. Our new method consists of training a neural network to emulate one or more physics schemes and then, instead of coding tangent-linear and adjoint models of the physics scheme manually or automatically, using the tangent-linear and adjoint versions of the neural network emulator. The neural network code is significantly easier to linearise compared with a normal physical parametrization scheme, owing to the inherently trivial differentiability of neural networks and their regular structure. Furthermore, the same code can in principle be reused for the tangent-linear and adjoint versions of multiple model components, provided that
the nonlinear counterparts can all be accurately emulated by the same neural network architecture. This also applies if one of those model components is changed — the emulator could simply be retrained to emulate the new scheme and the resulting parameters reused in the original tangent-linear and adjoint neural networks, without the need for further coding. Here we test this idea with the physics scheme that describes how gravity waves generated by non-orographic processes (e.g. tropospheric convection) travel to the stratosphere and break, thereby acting as a drag on the atmospheric flow.

A number of previous studies have successfully attempted to represent the effects of unresolved physical processes through neural networks (e.g. Chevallier et al. (1998); V. M. Krasnopolsky et al. (2005); Rasp et al. (2018); Brenowitz and Bretherton (2018); Yuval et al. (2021)) and we refer the reader to V. M. Krasnopolsky (2020) for a summary of the state-of-the-art. Some studies went further in quantifying the accuracy of neural networks by inspecting their Jacobians, i.e. the matrix of first derivatives of all output variables with respect to all input variables. This is relevant to our effort here as the Jacobian and its transpose are required to build the tangent-linear and adjoint models, respectively, of the neural network. Aires et al. (1999, 2004) demonstrated that an otherwise accurate neural network can have a noisy Jacobian. This is because the cost functions typically employed when training neural networks do not explicitly encourage neural networks with a smooth first derivative. They proposed to adopt certain regularisation constraints during the training step to alleviate this problem. Chevallier and Mahfouf (2001) found a similar issue with their own neural network emulation of the ECMWF radiation scheme, described by Chevallier et al. (2000). To build the tangent-linear and adjoint models of this scheme they used pre-computed mean Jacobian matrices. This approach was used operationally at ECMWF for a time (Janisková et al., 2002). Finally, V. Krasnopolsky (2006) proposed to minimise noise in the Jacobians by averaging the Jacobian across an “ensemble” of neural network-derived Jacobians. The subject of neural network Jacobians is therefore well established in the literature. However, there are comparatively few recent studies on this key feature of neural network-based parametrisation schemes. Furthermore, to the best of our knowledge, our use of flow-dependent neural network Jacobians (as opposed to the pre-computed Jacobians of Chevallier and Mahfouf (2001)) to build tangent-linear and adjoint models for data assimilation is novel.

In Section 2 we describe the chosen physics scheme and how we emulate it with a neural network, in Section 3 we describe how to construct the tangent-linear and adjoint
models of a neural network, in Section 4 we explain the standard procedure for testing
tangent-linear and adjoint model changes at ECMWF and apply these procedures to our
neural network, in Section 5 we show results from data assimilation and weather fore-
cast experiments using the neural network, and finally we conclude in Section 6.

2 Neural network emulation of the non-orographic gravity wave drag
scheme

The physical parametrization scheme that we focus on here describes how gravity
waves generated in the troposphere by non-orographic processes, such as convection and
frontogenesis, travel to the middle atmosphere and break, thereby acting as a drag on
the resolved flow. This phenomenon is known to influence the dynamics of the strato-
sphere and the levels above, including the Brewer-Dobson circulation and the quasi-biennial
oscillation. Since model cycle 35r3 of the ECMWF model, the Integrated Forecasting
System (IFS), went into operations in 2009, the effects of nonographic gravity wave drag
have been parametrized (Orr et al., 2010), based on the scheme of Scinocca (2003). This
model upgrade delivered a much improved representation of the middle atmosphere, com-
pared with the previous scheme based on a crude Rayleigh friction term.

In this study we chose the non-orographic gravity wave drag scheme as our case
study as we had prior experience in neural network emulation for this scheme. We de-
veloped a neural network-based emulator which in principle can be used as a drop-in re-
placement for the original scheme. Our companion paper, Chantry et al. (2021), focuses
on the development of the emulator, including the network design, training data gener-
ation and training procedure, and its testing through forecast and climatology verifica-
tion. Here we summarise the relevant details of the emulation.

The neural network architecture we chose is the fully-connected multilayer percep-
tron, which is arguably the simplest architecture which can be used for deep learning.
Viewed generally, this class of neural network can be written as a vector-valued function
of a vector that maps an input $x$ to an output $y$:

$$y = \mathbf{F}(x)$$

In this study we deal exclusively with multilayered networks of $N-1$ nonlinear layers
and 1 linear output layer, with each nonlinear layer (i.e. each of the input and hidden
layers) having a constant width. If $\mathbf{W}_i$ is the matrix containing the weights of the $i$th
layer (where \(i = 0\) indicates the input layer), \(b_i\) is the vector containing the biases in
the \(i\)th layer and \(h\) is the vector-valued function of a vector that evaluates the activa-
tion function for each element of its input, then the operation of the neural network up
to the \(i\)th layer can be written recursively as

\[
y_i = \begin{cases} 
W_0 x + b_0 & i = 0 \\
W_i h(y_{i-1}) + b_i & 0 < i \leq N 
\end{cases}
\]  

(2)

By this definition \(y = y_N\). Note that it is not conventional to define the partial out-
put of a neural network, \(y_i\), as including the application of layer \(i\)'s weights and biases,
as we have done here. Usually \(y_i\) is defined as the immediate output of the activation
function. We use this formulation to allow brevity in the equivalent tangent-linear and
adjoint expressions given below.

In this study, we propose to replace one or more of the physical parametrization
schemes used in a numerical model of the weather with a neural network. Under this scheme,
the function \(F\) in equation 1 would map a vector of variables describing the atmospheric
state, \(x\), to a vector containing the contributions to the tendencies (i.e. time derivatives)
of variables, \(y\), from the physical process encoded in \(F\). For the specific case study of non-
orographic gravity wave drag, \(x\) contains vertical columns of zonal and meridional wind
velocities and temperature along with pressure and geopotential at the surface. Then,
\(y\) contains the contribution to the tendencies of zonal and meridional wind velocities from
non-orographic gravity wave drag. Note that this parametrization scheme acts on each
atmospheric column independently and in isolation. It can therefore be perfectly par-
allelised in the horizontal directions.

We used the hyperbolic tangent as the activation function for all layers as its deriva-
tive can be computed easily and is completely smooth. We manually tuned the layer width
(keeping a constant width for each layer except the output layer) and the number of lay-
ers of the network, aiming to minimise the root-mean-square error over the training data
set, while keeping the total number of degrees-of-freedom fixed. Ultimately we settled
on a network of 6 hidden layers (so that the total number of layers including the out-
put layer, \(N\), is 7) each with a width of 43 neurons. For each variable with a vertical di-
mension, we considered only the top \(N_{\text{lev}} = 93\) out of a total of 137 vertical model lev-
els, as the parametrization scheme is only active for these levels. The input layer there-
fore has a width of \(3 \times 93 + 1 + 1 = 281\) elements and the output layer has a width of
2 × 93 = 186 elements. Note that the model considered here is comprised of 137 vertical levels, whereas the model considered by our companion paper, Chantry et al. (2021), is comprised of 91 vertical levels.

3 Linearizing neural networks

In order to use the neural network in 4D-Var data assimilation, as described in Section 1, we require its tangent-linear and adjoint. In other words, we also require linear functions $F$ and $F^\top$, defined by

$$\delta y = F(x, \delta x)$$

and

$$\frac{\partial J}{\partial x} = F^\top(x, \frac{\partial J}{\partial y}),$$

where $\delta x$ and $\delta y$ are perturbations to $x$ and $y$, respectively, and $J$ is an arbitrary scalar function of the output of the neural network, $y$.

The tangent-linear of the neural network can be found by taking the derivative of equation 2 and using the chain rule. Before we present this, we first define two useful functions. Firstly, $h'$ is the vector-valued function of a vector that evaluates the derivative of the activation function for each element of its input. Secondly,

$$G(z) = \text{diag}(h'(z)).$$

Then, we can write an explicit form for equation 3, the tangent-linear of the neural network:

$$\delta y = W_N \left[ \prod_{i=0}^{N-1} G(y_i(x)) W_i \right] \delta x.$$

Note that left-multiplying the matrix $G$ with a vector is equivalent to an element-wise vector product between the diagonal of $G$ and the right vector. We write $G$ as a matrix for notational simplicity.

An explicit form for equation 4, the adjoint of the neural network, can be found by taking the transpose of equation 6:

$$\frac{\partial J}{\partial x} = \left[ \prod_{i=0}^{N-1} W_i^\top G^\top(y_i(x)) \right] W_N^\top \frac{\partial J}{\partial y}.$$
As in Chantry et al. (2021), we trained our neural network using TensorFlow, in Python, but integrated the network back into the IFS by writing our own Fortran module. This module reads the network parameters generated by TensorFlow and executes the neural network on the inputs provided by the IFS, acting as a drop-in replacement for the existing non-orographic gravity wave drag scheme. This allowed us to subject our neural network code to the standard tangent-linear and adjoint model evaluation for model changes in the IFS.

4 Tangent-linear and adjoint model evaluation

When evaluating a model change at ECMWF we must always verify that the nonlinear, tangent-linear and adjoint models remain consistent with each other. If there is any inconsistency between the tangent-linear and adjoint models due to a code bug, for example, then this can be expected to disrupt the convergence of the 4D-Var cost function minimisation. In our case we are essentially proposing an alternative scheme to be used for non-orographic gravity wave drag and so we follow the same procedure in testing the linearised models that we would follow for any other model change.

We began by evaluating the neural network subroutines in isolation, namely those that are intended to replace the existing non-orographic gravity wave drag subroutines. These subroutines, the nonlinear, tangent-linear and adjoint versions of the neural network, must satisfy principally two tests before we can consider integrating them back into the IFS itself. As before, if the nonlinear, tangent-linear and adjoint subroutines are denoted by $\mathcal{F}$, $\mathbf{F}$ and $\mathbf{F}^\top$, respectively, then firstly $\mathcal{F}$ and $\mathbf{F}$ must satisfy the expression

$$\lim_{\alpha \to 0} [\mathcal{F} (x_0 + \alpha \delta x) - \mathcal{F} (x_0) - \mathbf{F} (x_0) \alpha \delta x] = 0,$$

where $x_0$ is an arbitrary point about which the linearisation is performed, $\alpha$ is a number that tends to zero and $\delta x$ is an arbitrary vector. This test essentially measures the accuracy of the tangent-linear model as a first-order Taylor approximation to the nonlinear model. In practice this test is performed by fixing $x$ and $\delta x$ and evaluating the subtraction in equation 8 for smaller and smaller values of $\alpha$ and verifying that the result of the subtraction indeed approaches zero. We verified that this was the case.

The second test performed against the subroutines in isolation was to verify that the adjoint identity is satisfied, namely that the expression

$$[\mathbf{F} (x_0) \delta x]^\top \delta y = \delta x^\top [\mathbf{F}^\top (x_0) \delta y]$$

(9)
holds, where $\delta y$ is an arbitrary vector. In fact, given that the operators $F$ and $F^\top$ are actually computer subroutines employing floating-point arithmetic, the identity in equation 9 will not hold exactly but instead be violated somewhat (Hatfield et al., 2020). When using double-precision the left-hand-side and right-hand-side can correspond to up to about 16 decimal digits. We verified that this was the case, observing no difference between the left-hand- and right-hand-sides of equation 9 up to the machine epsilon. The code for these two tests is available online (Hatfield & Chantry, 2021). Having verified that our neural networks pass the tangent-linear and adjoint tests at the individual subroutine level, we then applied similar tests at the level of the entire IFS model. These tests are far more stringent and provide a strong indication as to whether the code can be used to perform data assimilation or not.

### 4.1 Tangent-linear test

If the nonlinear model operator that evolves a model state forward over a certain time period (usually 12 hours in the case of 4D-Var) is given by $M$ then the tangent-linear model is given by $M$. This tangent-linear test compares a perturbation $\delta x$ evolved by the tangent-linear model linearised about a reference state $x_0$ against the difference between the two states evolved by the nonlinear model initialised at $x_0$ and $x_0+\delta x$. That is, we compute

$$
\epsilon = \langle |M(x_0 + \delta x) - M(x_0) - M(x_0)\delta x| \rangle,
$$

where the angle brackets denote a spatial average (global or zonal, for example) and the vertical bars represent taking the absolute value. This test is essentially a model-wide version of the isolated subroutine tangent-linear test of equation 8, performed using perturbations with a magnitude comparable to actual analysis increments that the tangent-linear model will encounter when used to perform 4D-Var data assimilation. The temperature variables in $\delta x$ here have magnitudes of several Kelvin, for example. Note also that this test does not consider the physical accuracy of the nonlinear or tangent-linear models. Even a physically inaccurate nonlinear and tangent-linear model pair can pass this test, as long as they correspond closely according to equation 10.

The value of $\epsilon$ in general will not be zero, even when the tangent-linear model is coded perfectly, as atmospheric evolution is often nonlinear and so violates the first-order Taylor approximation, especially for the relatively large values of $\delta x$ we consider here.
Instead we aim to show that the $\epsilon$ computed using the neural network non-orographic gravity wave drag scheme (labelled NN) is comparable to the $\epsilon$ computed from a reference (labelled REF), namely the operational configuration of the IFS. Specifically, we look at the relative change in $\epsilon$ when switching from the reference model to the neural network model, $\epsilon_{\text{rel}} = (\epsilon_{\text{NN}} - \epsilon_{\text{REF}})/\epsilon_{\text{REF}}$. We use the same NN and REF labels for all following experiments in the paper.

Figure 1. The relative change in horizontally-averaged $\epsilon$ when using the neural network in both the tangent-linear and nonlinear model (the “easy” test) and just the tangent-linear model (the “hard” test), for three variables across all model levels. The tests were run for 20 different start dates. The central lines and shaded areas represent the mean and central 80%, respectively, of the entire distribution. Negative values indicate that the tangent-linear/nonlinear correspondence is better for the neural network than the reference, the operational IFS.

We present the results of the tangent-linear test for two cases, which we call the “easy test” and the “hard test”. For the easy test, both the nonlinear model $\mathcal{M}$ and the tangent-linear model $\mathbf{M}$ use the neural network for the non-orographic gravity wave drag scheme. For the hard test, we use the neural network for $\mathbf{M}$ but the original physics scheme for $\mathcal{M}$. The latter test is relevant for the weather forecast tests we present later, in which we only use the neural network for the tangent-linear and adjoint model, not the nonlinear model. It is a harder test, as the tangent-linear neural network is tested against the original nonlinear non-orographic gravity wave drag scheme, not the nonlinear neu-
ral network as in the easy test. This test therefore also indirectly measures the accuracy of the neural network with respect to the original physics scheme.

The results for both the easy and hard tangent-linear tests are shown in Figure 1. These tests were run over 20 start dates. For the distribution over these start dates, the central lines and shaded areas in Figure 1 represent the mean and central 80%, respectively. For the easy test, $\epsilon_{rel}$ is around zero for most of the atmosphere, except for the mesosphere where it is actually negative. This indicates that, in this region, the nonlinear and tangent-linear models actually correspond more closely for NN compared with REF. This could be due to the nature of the tangent-linear model used by REF, which is based on that of the operational IFS. This setup uses a regularized version of the non-orographic gravity wave drag scheme for the tangent-linear case in which, for example, momentum fluxes for high phase speed spectral components are zeroed. This helps to achieve numerical stability in the 4D-Var minimisation (Janisková & Lopez, 2013). The tangent-linear model for NN, on the other hand, is derived directly from an emulation of the nonlinear scheme without any of these regularizations. Hence, the formulations of the nonlinear and tangent-linear model are closer for NN and this likely explains the negative values of $\epsilon_{rel}$.

For the hard test, $\epsilon_{rel}$ is again mostly close to zero, but is positive in the mesosphere, indicating that the nonlinear and tangent-linear models correspond less well for NN than for REF here. For the hard test, the nonlinear and tangent-linear models for NN are formulated differently. Clearly this difference is bigger for NN than for REF, hence the positive values of $\epsilon_{rel}$. Through further training and hyperparameter exploration, $\epsilon_{rel}$ could probably be brought closer to zero. However, further effort spent to improve these results would ultimately not be useful and we are not concerned about the results. The tangent-linear model used for data assimilation at ECMWF runs at a lower resolution than the nonlinear model, to save computational resources. There is therefore always an unavoidable asymmetry between the nonlinear and tangent-linear models in practice. Having demonstrated that NN passes the tangent-linear test, we now move on to the adjoint test.
Figure 2. The results from the adjoint test applied to NN. The numbers in the boxes indicate the number of consecutive equal digits between the left- and right-hand-sides of equation 11. This test is applied across 20 dates and for 10 different configurations of the linearized physics, in which different physics schemes are switched on and off.

4.2 Adjoint test

If the tangent-linear model is given by $M$ then the adjoint model is given by $M^\top$ such that the expression

$$[M(x_0) \delta x]^\top \delta y = \delta x^\top [M^\top(x_0) \delta y]$$

(11)

holds, where $\delta y$ is another arbitrary model state perturbation. This is equivalent to the isolated subroutine test of equation 9, though in practice we consider a correspondence of at least 10 consecutive equal digits between the left-hand- and right-hand-sides of equation 11 to be a “pass”. A greater mismatch than this can occur if the numerical precision is too low (single-precision, for example) but it can also be evidence of an incorrectly coded tangent-linear or adjoint model.

The results from the adjoint test with experiment NN are shown in Figure 2. We ran this test across 20 different dates and for 10 different configurations of the model. Each date and model configuration pair produces a single number — the number of consecutive matching digits between the inner products on the left-hand- and right-hand-sides of equation 11. We do not detail all 10 of these configurations here, but essentially the first two configurations include the full physical parametrization suite of the IFS with slightly different options whereas the other configurations have different parametrization schemes switched on and off to help in debugging the tangent-linear and adjoint code. Across most dates and physics configurations there is a good tangent-linear and adjoint
correspondence with at least 12 consecutively equal digits. There are a number of cases with a lower correspondence, including two cases with 9 digits. However, such cases typically occur even with the operational IFS, so we do not investigate them any further. We are therefore confident that the adjoint of the neural network is also coded correctly and move on to the data assimilation and weather forecasting experiments.

5 Data assimilation and weather forecasting experiments

Though we know from experience that model changes need to pass the tests in Section 4 to be used in 4D-Var data assimilation, these tests alone are not sufficient evidence that 4D-Var can actually be performed successfully. Here we demonstrate the suitability of our neural network-based tangent-linear and adjoint models for data assimilation by performing an actual cycled 4D-Var data assimilation experiment. Our experiment covered the Winter period from December 1st 2018 to February 28th 2019. In this experiment consecutive 4D-Var data assimilation cycles were run every 12 hours and 10 day forecasts were issued from the final analysis of each assimilation cycle. In total, 177 forecasts were performed. The horizontal resolution of the nonlinear integrations (the 4D-Var trajectory and the actual forecasts) was TCo399 (a triangular truncation with a maximum total and zonal wavenumber of 399 in spectral space with a cubic-octahedral reduced Gaussian grid in grid point space) which roughly corresponds to a grid-spacing of 25 km at the Equator. The horizontal resolution of the tangent-linear and adjoint model integrations was TL95 (a triangular truncation of 95 with a linear reduced Gaussian grid), TL159 and TL255 for the first, second and third inner loops of the 4D-Var minimisation, respectively. The model had 137 vertical levels for all integrations. For reference, the high-resolution operational configuration of the IFS runs at a resolution of TCo1279 (roughly 9 km grid-spacing at the Equator) for the 4D-Var trajectories and has four inner loops at resolutions of TL255, TL319, TL399 and TL399, respectively. There are 137 vertical levels in all cases. Our experiments were performed with the IFS at model cycle 46r1.

We evaluate the performance of the neural network based on the quality of the weather forecasts initialised from the analysis products of the data assimilation cycle. This is done by reference to the operational configuration of the IFS, which uses hand-coded tangent-linear and adjoint models of the non-orographic gravity wave drag scheme. We also present diagnostics evaluating how well the model of experiment NN fits observations, compared with REF. As mentioned earlier when explaining the hard tangent-linear test, we only
use the neural network in the tangent-linear and adjoint models, not in the nonlinear model used to generate the 4D-Var trajectories and the actual forecasts. This is so that any changes in forecast scores between NN and REF can be attributed only to the difference in tangent-linear and adjoint models. A detailed verification of the nonlinear neural network emulator can be found in our companion paper, Chantry et al. (2021).

**Figure 3.** The relative difference in root-mean-square error of temperature of experiment NN compared with experiment REF (blue indicates that NN is better than REF) averaged in the zonal direction and across all forecasts in the experimental period, for a number of forecast lead times (indicated in each subfigure title by “T+” with the number of hours after the initial time). The relative difference is computed by dividing by the error of REF. Hatching indicates that differences are significant with 95% confidence.

Our headline result is shown in Figure 3. This shows the relative difference in root-mean-square error of forecasted temperature averaged in the zonal direction and across all forecasts in the experimental period, for a number of forecast lead times. The differ-
ence in error shown is between the two experiments NN and REF, with red or blue colours indicating that NN is worse or better than REF, respectively. For both experiments, the forecast error is computed with respect to the experiment’s own analysis (rather than an independent analysis, for example). The absolute value of the differences amount to no more than 4%, though this is neither decisively in the positive or negative direction and in any case is not statistically significant, which is demonstrated by the scarcity of significance hatching. In other words, experiment NN does not perform statistically significantly differently from experiment REF. We also looked at the fields of geopotential height, relative humidity, zonal wind and meridional wind and there too the conclusion is the same (which we don’t show here).

![Relative change in RMS of ATMS observation departures between experiments NN and REF (%)](image)

**Figure 4.** The relative change in root-mean-square error of Advanced Technology Microwave Sounder (ATMS) observational departures of experiment NN with respect to experiment REF. The analysis and background departures are both shown. Negative values indicate that NN performs better than REF, and vice versa for positive values.

In Figure 4 we compare both the analysis and background (first guess of the atmospheric state at the beginning of the 4D-Var process) for each experiment with observations taken from the Advanced Technology Microwave Sounder (ATMS) satellite instrument. We compute the root-mean-square error between the observations and the analysis and then the relative change of this error between experiments REF and NN. We then repeat this procedure but for the background instead of the analysis. This is averaged across the whole globe and all data assimilation cycles. Confidence intervals
of 95% are also shown, computed according to the procedure in Geer (2016). Note that channels 16 and 17 are missing from these figures as those channels are not assimilated at ECMWF.

For almost all channels there is no significant difference between the two experiments according to the root-mean-square of their analysis and background departures. There are several channels at which the change in departure root-mean-square is outside of the 95% confidence interval, such as channels 8, 14 and 15 for the analysis departures and channels 8, 9 and 10 for the background departures. However, the confidence intervals computed through the ECMWF verification software do not include factors to account for temporal autocorrelation which could inflate these intervals by up to 30%. We are therefore not concerned about these apparently statistically significant differences.

We also considered observations from another microwave sounder, the Advanced Microwave Sounding Unit A (AMSU-A), but we could not detect any significant changes from experiment NN compared with experiment REF there either.

**Figure 5.** The relative change in root-mean-square of Global Positioning System radio occultation (GPSRO) observational departures of experiment NN with respect to experiment REF. The analysis and background departures are both shown. Negative values indicate that NN performs better than REF, and vice versa for positive values.

Finally, we show an equivalent plot to Figure 4 but for Global Positioning System (GPS) radio occultation observations in Figure 5. These observations measure the bending of radio waves from GPS satellites as they pass through the atmosphere to be inter-
cepted by satellites in low Earth orbit. They are given as a function of the “tangent height”,
the lowest altitude that the wave reaches as it passes from source to receiver. As in Fig-
ure 5 there is almost no discernible difference between experiments NN and REF, as the
relative change in RMS is not significantly different from 0% at most tangent heights.
The only exception is around 2 km where there is an apparent degradation for experi-
iment NN. At this level experiment NN permitted a slightly larger number of observa-
tions to pass quality control, compared with REF. These observations happened to match
more poorly against analysis and background than at other tangent heights. In any case,
GPS radio occultation observations are most reliable between 10 - 30 km tangent height
where there is no discernible change in RMS error so we are not concerned about the ap-
parent difference at 2 km.

6 Conclusion

Techniques from the field of machine learning have a number of potential applica-
tions within NWP. Here we have demonstrated one use for neural networks in variational
data assimilation that takes advantage of their easy differentiability. We began by train-
ing an emulator for a physical parametrization scheme representing drag in the middle
atmosphere due to the breaking of gravity waves forced by non-orographic processes. Then,
we developed tangent-linear and adjoint versions of this neural network and implemented
these in an operational NWP model. We then verified that our neural network-based parametra-
tion scheme passes the standard internal testing procedure for tangent-linear and adjoint
model changes at ECMWF and that it can also be used successfully in a data assim-
ilation and weather forecast experiment. Our experiment using a neural network gave no
statistically significant difference to the forecast scores, especially near analysis time, com-
pared with a reference that did not use a neural network.

If this approach could be extended to other physics parametrization schemes as well
then it holds the potential to significantly simplify tangent-linear and adjoint model main-
tenance not just at ECMWF but other forecasting centres around the world that use 4D-
Var. The evaluation of the neural network is nothing more than a large batch of matrix-
matrix multiplications, with the activation function evaluated after each hidden layer.
It is therefore ideal for use on hardware optimised for such machine learning purposes,
like graphics processing units. Therefore, thanks to improvements in hardware porta-
bility, the technique outlined here may help to keep 4D-Var competitive as we enter the
age of heterogeneous supercomputer architectures. Furthermore, as demonstrated here, the neural network can be trained to emulate the full-complexity nonlinear model component, with the weights being reused in the tangent-linear and adjoint versions. This is unlike the existing approach in which physical processes must occasionally be simplified for reasons of computational cost. The formulation of the linearised physics can therefore be kept closer to that of the nonlinear physics, and in some cases could include physical processes usually omitted.

As mentioned earlier, another motivation for using simplified physics and additional regularisation procedures in tangent-linear and adjoint models is to improve the operational stability of these models. The perturbations and gradients (the inputs to the tangent-linear and adjoint models, respectively) actually encountered in operational data assimilation are far from infinitesimal in size, and therefore the tangent-linear and adjoint models must be modified (“regularised”) to handle these without producing unphysical outputs (Janisková & Lopez, 2013). Furthermore, some physical processes are modelled by discontinuous functions, with the derivative ill-defined at the threshold. In principle a neural network emulator would suffer from the same problems. A step function would be automatically smoothed when emulated by a neural network, so in that case the neural network would at least remain continuously differentiable. However, the issue of large gradients would remain which could render the tangent-linear and adjoint models unstable in cases of strong nonlinearity, just as for the conventional paradigm of manually linearised physics. We did not encounter any issues in our experiments despite having trained the neural network on the unregularised non-orographic gravity wave drag scheme. However, a similar emulation procedure applied to the cloud scheme may present problems. In the existing cloud scheme, a manual intervention in the tangent-linear and adjoint models drawing on domain knowledge of cloud processes was required to ensure stability (Tompkins & Janisková, 2004). If such an intervention is also necessary for the linearisation of the neural network emulator then this arguably defeats the purpose of such an automated approach. If problems appear, it may be sufficient to adopt a simple regularising constraint during the training of the neural network. An additional term in the objective function that penalises large gradients would be one solution, though the regularisation strength would constitute a further hyperparameter to be tuned. Further work on other physical parametrization schemes within the IFS, including the convection and large-scale cloud schemes, will help to answer these questions.
Going forward, in addition to giving more attention to the matter of regularisation, we intend to identify other benefits to the neural network approach. As mentioned, the tangent-linear and adjoint models of the neural network need only be written once. They can then in principle be reused for other model components provided that accurate neural network emulators of those components can be trained. Other physical parametrization schemes could be considered individually or they could be grouped together. For example, both the orographic and non-orographic gravity wave drag schemes could be emulated together. This would provide a simplification of the existing tangent-linear and adjoint code and also a computational acceleration, assuming that the emulation could still be performed with a relatively cheap neural network of the kind employed in this paper. The computational acceleration provided by the neural network is itself another avenue to explore. We did not discuss this here as the cost of the non-orographic gravity wave drag scheme is in any case only around 1% of the total cost of the model evaluation. However, given that the physical parametrizations as a whole are often 25% of the total cost of a model integration (Bauer et al., 2020), the computational cost savings from neural network emulators could prove to be their main asset.

The idea that we have promoted in this paper, that of automatic generation of tangent-linear and adjoint models by neural networks, demonstrates that machine learning techniques provide auxiliary benefits besides simple replacement of model components by emulators. We believe that neural networks could be an essential tool for keeping forecasting and data assimilation methods based on tangent-linear and adjoint models competitive, especially as we enter the exascale era.

**Acknowledgments**

Many thanks are owed to Michail Diamantakis for finding bugs in the data assimilation experiment configuration and Inna Polichtchouk and Sean Healy for help with analysis of the weather forecasting experiments. P. Düben gratefully acknowledges funding from the Royal Society for his University Research Fellowship as well as the MAELSTROM and AI4Copernicus EU projects (grant agreement No 955513 and 101016798). S. Hatfield and P. Düben have also received funding from the ESIWACE EU project (grant agreement No 823988). M. Chantry was supported by a grant from the Office of Naval Research Global. The data used to train the neural networks for 137 levels (as opposed to the 91 level neural networks in our companion paper, (Chantry et al., 2021)) are avail-
The code for the neural network, including the tangent-linear and adjoint versions, is available through (Hatfield & Chantry, 2021).

References

Aires, F., Pringent, C., & Rossow, W. B. (2004). Neural network uncertainty assessment using Bayesian statistics with application to remote sensing: 3. Network Jacobians. *Journal of Geophysical Research D: Atmospheres, 109*(10), 1–14. doi: 10.1029/2003JD004175

Aires, F., Schmitt, M., Chedin, A., & Scott, N. (1999). The "weight smoothing" regularization of MLP for Jacobian stabilization. *IEEE Transactions on Neural Networks, 10*(6), 1502–1510. doi: 10.1109/72.809096

Bauer, P., Quintino, T., Wedi, N., Bonanni, A., Chrust, M., Deconinck, W., ... Zanna, C. (2020). *The ECMWF Scalability Programme: Progress and Plans* (Tech. Rep.). doi: 10.21957/gdit22ulm

Bonavita, M., Hamrud, M., & Isaksen, L. (2015). EnKF and Hybrid Gain Ensemble Data Assimilation. Part II: EnKF and Hybrid Gain Results. *Monthly Weather Review, 143*(12), 4865–4882. Retrieved from http://journals.ametsoc.org/doi/abs/10.1175/MWR-D-15-0071.1 doi: 10.1175/MWR-D-15-0071.1

Brenowitz, N. D., & Bretherton, C. S. (2018). Prognostic Validation of a Neural Network Unified Physics Parameterization. *Geophysical Research Letters, 45*(12), 6289–6298. doi: 10.1029/2018GL078510

Chantray, M., Hatfield, S., Duben, P., Polichtchouk, I., & Palmer, T. (2021). Machine learning emulation of gravity wave drag in numerical weather forecasting. , 1–23. Retrieved from http://arxiv.org/abs/2101.08195

Chevallier, F., Chéruy, F., Scott, N. A., & Chédin, A. (1998, nov). A Neural Network Approach for a Fast and Accurate Computation of a Longwave Radiative Budget. *Journal of Applied Meteorology, 37*(11), 1385–1397. Retrieved from http://journals.ametsoc.org/doi/abs/10.1175/1520-0450(1998)037<1385:ANNAFA>2.0.CO;2 281998\%203C1385\%3E\%3A\%3E2.0.CO;2

Chevallier, F., & Mahfouf, J. F. (2001). Evaluation of the Jacobians of infrared radiation models for variational data assimilation. *Journal of Applied Meteorology,
Chevallier, F., Morcrette, J. J., Chéruy, F., & Scott, N. A. (2000). Use of a neural-network-based long-wave radiative-transfer scheme in the ECMWF atmospheric model. *Quarterly Journal of the Royal Meteorological Society, 126*(563), 761–776. doi: 10.1002/qj.49712656318

Courtier, P., Thépaut, J.-N., & Hollingsworth, A. (1994). A strategy for operational implementation of 4D-Var, using an incremental approach. *Quart. J. Roy. Meteor. Soc.*, 120(519), 1367–1387. Retrieved from http://dx.doi.org/10.1002/qj.49712051912 doi: 10.1002/qj.49712051912

Geer, A. J. (2016). Significance of changes in medium-range forecast scores. *Tellus, Series A: Dynamic Meteorology and Oceanography, 68*(1). doi: 10.3402/tellusa.v68.30229

Geer, A. J. (2021, apr). Learning earth system models from observations: machine learning or data assimilation? *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 379*(2194), 20200089. Retrieved from https://royalsocietypublishing.org/doi/10.1098/rsta.2020.0089 doi: 10.1098/rsta.2020.0089

Hamrud, M., Bonavita, M., & Isaksen, L. (2015). EnKF and Hybrid Gain Ensemble Data Assimilation. Part I: EnKF Implementation. *Monthly Weather Review, 143*(12), 4847–4864. Retrieved from http://journals.ametsoc.org/abs/10.1175/MWR-D-15-0071.1 doi: 10.1175/MWR-D-15-0071.1

Hatfield, S., & Chantry, M. (2021). *samhatfield/nn_tl_ad: Publication*. doi: 10.5281/zenodo.4525830

Hatfield, S., McRae, A., Palmer, T., & Düben, P. (2020). Single-precision in the tangent-linear and adjoint models of incremental 4D-Var. *Monthly Weather Review, 1541–1552*. doi: 10.1175/mwr-d-19-0291.1

Janisková, M., & Lopez, P. (2013). Linearized Physics for Data Assimilation at ECMWF. In *Data assimilation for atmospheric, oceanic and hydrologic applications (vol. ii)* (Vol. II, pp. 251–286). Berlin, Heidelberg: Springer Berlin Heidelberg. Retrieved from http://link.springer.com/10.1007/978-3-642-35088-7_11 doi: 10.1007/978-3-642-35088-7_11

Janisková, M., Mahfouf, J.-F., Morcrette, J.-J., & Chevallier, F. (2002). *Linearized
radiation and cloud schemes in the ECMWF model: Development and evaluation (Vol. 128; Tech. Rep.).

Krasnopolsky, V. (2006). Reducing uncertainties in neural network jacobians and improving accuracy of neural network emulations with NN ensemble approaches. In IEEE International Conference on Neural Networks - Conference Proceedings (pp. 4587–4594). Institute of Electrical and Electronics Engineers Inc. doi: 10.1109/ijcnn.2006.247087

Krasnopolsky, V. M. (2020, Feb). Using Machine Learning for Model Physics: an Overview. ArXiv (January). Retrieved from http://arxiv.org/abs/2002.00416

Krasnopolsky, V. M., Fox-Rabinovitz, M. S., & Chalikov, D. V. (2005, May). New Approach to Calculation of Atmospheric Model Physics: Accurate and Fast Neural Network Emulation of Longwave Radiation in a Climate Model. Monthly Weather Review, 133(5), 1370–1383. Retrieved from http://journals.ametsoc.org/doi/abs/10.1175/2009MWR2923.1 doi: 10.1175/MWR2923.1

Orr, A., Bechtold, P., Scinocca, J., Ern, M., & Janiskova, M. (2010). Improved middle atmosphere climate and forecasts in the ECMWF model through a nonorographic gravity wave drag parameterization. Journal of Climate, 23(22), 5905–5926. doi: 10.1175/2010JCLI3490.1

Payne, T. J. (2021, Jan). A Hybrid Differential-Ensemble Linear Forecast Model for 4D-Var. Monthly Weather Review, 149(1), 3–19. Retrieved from https://journals.ametsoc.org/view/journals/mwre/149/1/mwr-d-20-0088.1.xml doi: 10.1175/MWR-D-20-0088.1

Rabier, F. (1998). Extended assimilation and forecast experiments with a four-dimensional variational assimilation system. Quarterly Journal of the Royal Meteorological Society, 124(550), 1861–1887. doi: 10.1256/smsqj.55004

Rasp, S., Pritchard, M. S., & Gentile, P. (2018). Deep learning to represent subgrid processes in climate models. Proceedings of the National Academy of Sciences of the United States of America, 115(39), 9684–9689. doi: 10.1073/pnas.1810286115

Scinocca, J. F. (2003). An accurate spectral nonographic gravity wave drag parame-
terization for general circulation models. *Journal of the Atmospheric Sciences*, 60(4), 667–682. doi: 10.1175/1520-0469(2003)060<0667:AASNGW>2.0.CO;2

Tompkins, A. M., & Janisková, M. (2004, oct). A cloud scheme for data assimilation: Description and initial tests. *Quarterly Journal of the Royal Meteorological Society*, 130(602 PART A), 2495–2517. doi: 10.1256/qj.03.162

Yuval, J., O’Gorman, P. A., & Hill, C. N. (2021, feb). Use of neural networks for stable, accurate and physically consistent parameterization of subgrid atmospheric processes with good performance at reduced precision. *Geophysical Research Letters*. Retrieved from https://onlinelibrary.wiley.com/doi/10.1029/2020GL091363 doi: 10.1029/2020GL091363
Figure 1.
\[(\varepsilon_{\text{NN}} - \varepsilon_{\text{ref}})/\varepsilon_{\text{rel}} \%\]

**Temperature**

**Zonal wind**

**Meridional wind**

Mesosphere

Stratosphere

Troposphere

Model level

-20.0 0.0 20.0

Easy

Hard
| Physics configurations | Dates |
|------------------------|-------|
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 3 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 4 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 5 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 6 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 7 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 8 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 9 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 10 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
Figure 3.
Relative change in temperature RMSE for experiment NN with respect to experiment REF
Figure 4.
Relative change in RMS of ATMS observation departures between experiments NN and REF (%)

Analysis

Channel

Background
Relative change in RMS of GPS radio occultation observation departures between experiments NN and REF (%)