Proceedings of the 3rd International Symposium on Uncertainty Quantification and Stochastic Modeling
D.A. Rade; A. M. G. Lima (Editors), ABCM, Maresias, SP, Brazil, February 15-19, 2016

Multilayer Perceptron on data assimilation applied to FSU global model

Rosangela S. Cintra[1], **Haroldo F. de Campos Velho**[1], **and Steven Cocke**[2]

[1] National Institute for Space Research, S. José dos Campos, SP, Brazil - rosangela.cintra, haroldo@lac.inpe.br

[2] Center of Ocean-Atmospheric Prediction Studies, Florida State University, Tallahassee, FL, USA - scocke@fsu.edu

Abstract: Numerical weather prediction (NWP) uses atmospheric general circulation models (AGCMs) to predict weather based on current weather conditions. The atmosphere could not be completely described due to inherent uncertainty. These uncertainties limit forecast model accuracy to about five or six days into the future. The process of entering observation data into mathematical model to generate the accurate initial conditions is called data assimilation (DA). This paper shows the results of a DA technique using artificial neural networks (NN) applied to an AGCM used in Florida State University (FSU) in USA. The Local Ensemble Transform Kalman filter (LETKF), a version of Kalman filter with ensembles to represent the model uncertainties, is a traditional DA scheme. We use Multilayer Perceptron data assimilation (MLP-DA) with supervised training algorithm where NN receives input vectors with their corresponding response from LETKF initial conditions. These DA schemes are applied to FSU Global Spectral Model (FSUGSM), a multilevel spectral primitive equation model at resolution T63L27. This data assimilation experiment is based in synthetic observations: surface pressure and upper-air temperature. We use a NN self-configuration method to find the optimal NN parameters to configure the MLP-DA with: four input vector nodes and one output node for the analysis vector. The NNs were trained with data from each month of 2001, 2002, and 2003. The MLP-DA cycle is performed for January 2004. The numerical results demonstrate the effectiveness of the MLP-DA technique for atmospheric data assimilation, since the initial conditions have similar quality to LETKF. The reduced computational cost allows the inclusion of greater number of observations and new data sources and the use of high resolution of models.

Keywords: data assimilation, artificial neural networks, ensemble Kalman filter, multilayer perceptron

INTRODUCTION

Predictions are made from computer models of the atmosphere by integrating the Navier-Stokes equations for a three dimensional multi-constituent multi-phase rotation fluid, and coupled to representations of the ocean and land surface. Predictions are continually put to the test through the daily weather forecast. Model forecasts have limits to the predictability of the behavior of the atmosphere. It is because the chaotic dynamics are sensitive to the error in the initial state (Lorenz, 1960). The accuracy of weather forecasts is influenced by the ability to represent computationally the full equations of motion that governs the atmosphere, in addition to error in initial conditions. Poorly known parameters are a key source of uncertainty. Even if the initial condition is well defined, errors in the parameters will affect the accuracy of forecasts.

Data assimilation (DA) is the process by which measurements and model predictions are combined to obtain an accurate representation of the state of the modelled system as its initial condition. According Talagrand (2008), the purpose of assimilation is to reconstruct as accurately as possible the atmospheric or oceanic flow, using all available appropriate information. The analysis for atmospheric flow is based on observational data and a model of the physical system, with some background information on initial condition (Hólm, 2008). An important problem in atmospheric data assimilation lies in the large number of degrees of freedom of NWP models. Very large numerical dimensions are required: $10^7 - 10^9$ parameters to be estimated with 2×10^7 observations per 24-hour period. The large number of degrees of freedom of covariance matrices involved can prohibit the implementation of the best assimilation method known where there is a need for the forecast to be ready in a short amount of time (Talagrand, 2008).

Data Assimilation

The key concept is that to assume a tracking process as a data assimilation process. DA is a mathematical technique that enables the optimal use of model and observational resources and offers the potential to generate forecasts that are more accurate than using a model alone. It is most commonly used to produce initial conditions for state estimation: estimating model variables keeping the model parameters fixed. However, it is also possible to use data assimilation to provide estimates of uncertain model parameters. Data assimilation techniques have been employed in the context of atmospheric and oceanic prediction, environmental and hydrological prediction, and ionosphere dynamics for some years. There are many different types of data assimilation algorithm, each varying in formulation, complexity, optimality and suitability for practical application. A useful overview of some of the most common data assimilation methods used in meteorology and oceanography are given in the review articles by Ghil and Malanotte-Rizzoli (1991). Detailed mathematical formulations can be found in texts such as Daley (1991) and Kalnay (2003).

The Kalman filter (KF) (Kalman, 1960) is a theoretically attractive algorithm for the optimal estimation of atmospheric states. The Bayesian scheme is approached using ensembles of integrations of comprehensive weather prediction models, with explicit perturbations to both initial conditions and model formulation; the resulting ensemble of forecasts can be interpreted as a probabilistic prediction. The ensemble Kalman filter (EnKF) (Evensen, 1994) uses a probability density function associated with the initial condition, characterizing the Bayesian approaches (Daley, 1991). The EnKF represents the model error by an ensemble of estimates in state space, which are meant to sample the probability for the state of the system. The local ensemble Kalman filter (LEKF) (Ott et al., 2004) proposes the EnKF scheme restricted to small areas (local). The local ensemble transform Kalman filter (LETKF; (Hunt, Kostelich and Szunyogh, 2007)) is an EnKF-based algorithm designed to be efficient on parallel computing architectures by taking an advantage of independent local analyses of the EnKF. A number of studies have shown promise of the LETKF with wide applications including global and regional atmosphere and ocean models (e.g.: (Miyoshi and Yamane, 2007), (Cintra and Cocke, 2014)).

The application of artificial neural networks (NN) was suggested as a possible technique for data assimilation by Hsieh and Tang (1998) and Liaqat et al. (2003). Nowosad et al. (2000) implemented an NN as an approach for data assimilation evaluating multilayer perceptron (MLP). Later, this approach was improved by Harter and Campos Velho(2008), evaluating the performance of NN by radial basis function, and Elman and Jordan recurrent networks. Furtado (2008) applied MLPs to emulate the particle filter and the variational method for data assimilation for the Lorenz chaotic system. Cintra (2010) applied the MLP technique to emulate LETKF to the AGCM *Simplified Parameterizations PrimitivE-Equation Dynamics* (SPEEDY) model, using syntethic observations. Then, Furtado (2012) applied the MLP method to emulate the *Inverse Ocean Modeling* (IOM) system, simulating ocean data using with the shallow-water model and the linear wave equation model. The methods using Artificial Neural Networks have shown consistent results with all implementation.

This paper presents the approach based on using NN to mimic the LETKF scheme of DA. The experiments use synthetic observations (surface pressure and absolute temperatureat each 6 hours on a day) to perform both data assimilation schemes. The goal to use the NN approach is to achieve a better computational performance with similar quality for the prediction, i.e., an computational efficient process of atmospheric data assimilation (the analysis). The experiment was conducted using the Florida State University Global Spectral Model (FSUGSM),(Cocke and Larrow, 2000) which is a 3D global atmospheric model. The spatial resolution considered is T63L27 for the spectral method. The LETKF (Cintra and Cocke, 2014) analysis is used as target for training artificial neural network.

METHODOLOGY

Weather Tracking as Nonlinear Data Assimilation

Considering a general nonlinear system with an $n-$dimensional sate vector x and a $m-$dimensional observation vector y evolving according to

$$x_{k+1} = f(x_k, t_k) + w_k \tag{1}$$

$$y_k = h(x_k, t_k) + v_k \tag{2}$$

where w_k and v_k are Gaussian noise terms with covariance matrices Q and R respectively. An ensemble Kalman filter (EnKF) approximates a nonlinear system using a finite weighted ensemble (Kalnay, 2003). This ensemble is the propagated forward a single time step using the model f and observed with h. The mean of the resulting state ensemble is the *a priori* estimate x_k and the mean of the observed ensemble is the predicted observation y_k^-. Denoting the covariance matrices P_{k-1} and P_k^y of the resulting state and observed ensemble, and the cross-covariance P_k^{xy} between the state and observed ensembles, the equations

$$K_k = P_k^{xy}(P_{k-1})_{-1}, \tag{3}$$

$$P_k = P_{k-1} - P_k^{xy}(P_k^y)_{-1}P_k^{xy}, \tag{4}$$

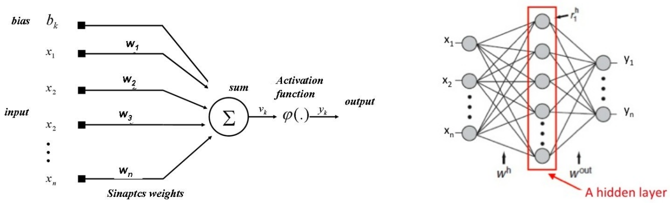
$$x_{k+1} = x_k + K_k(y_k - y_k^-. \tag{5}$$

update the state and covariance estimates using observations y_k.

In atmospheric data assimilation, after the current state by forecasting step, observations of the current and past state are combined by filtering step. Observations are information that we can obtain from various sources, mainly from satellites sensors. Forecasting is a step to predict a current state of a system from the last state by numerical weather prediction model. We consider this forecasting process as nonlinear process. Filtering step is a step to balance the predicted current state by forecasting step and current observations.

Artificial Neural Network

An artificial neural network (NN) is a computational system with parallel and distributed processing that has the ability to learn and store experimental knowledge. NN is composed of simple processing units that compute mathematical functions (usually nonlinear). NN consists of interconnected artificial neurons or nodes, which are inspired by biological neurons and their behaviour. The neurons are connected to others to form a network, which is used to model relationships between artificial neurons, see (Fig. 1(a)). The neuron processing can be nonlinear, parallel, local, and adaptable. Each



(a) (b)

Figure 1 – (a) Artificial neural network components. (b) Multylayer Perceptron.

artificial neuron is constituted by one or more inputs and outputs, and has a function to define outputs, associated with a learning rule. The connection between neurons stores a nonlinear weighted sum, called synaptic weight. In NN processing, the inputs are multiplied by weights; these results summarized then go through the activation function. This function activates or inhibits the next neuron. Mathematically, we can describe the i^{th} input with the following form:

$$\begin{array}{ll} \text{input summation:} & u_i = \sum_{j=1}^{p} w_{ij} x_j \\ \text{neuron output:} & y_i = \varphi(u_i) \end{array} \tag{6}$$

where $x_1, x_2, \cdots, x_n$ are the inputs; $w_{i1}, \cdots, w_{ip}$ are the synaptic weights; u_i is the output of linear combination; $\varphi(\cdot)$ is the activation function, and y_i is the i-th neuron output, n is number of patterns, p is number of layers. A feed-forward network, which processes in one direction from input to output, has a layered structure. The first layer of an NN is called the input layer, the intermediary layers are called hidden layers, and the last layer is called the output layer. The number of layers and the quantity of neurons in each is determined by the nature of the problem.

NN architectures are dependent upon the learning strategy adopted (Haykin, 2001). The multilayer perceptron (MLP) is the NN architecture used in this study; which the interconnections between the inputs and the output layer have at least one intermediate layer of neurons, a hidden layer (Haykin, 2007), see (Fig. 1(b)). NNs can solve nonlinear problems if nonlinear activation functions are used for the hidden and/or the output layers.

There are two distinct phases in using an MLP: the training phase (learning process) and the run phase (activation process). The training phase of the NN consists of an iterative process for adjusting the weights for the best performance of the NN in establishing the mapping of input and target vector pairs. The learning algorithm is a set of procedures for adjusting the weights. The goal is to minimize the error between the actual output y_i and the target output (d_i) of the training data. For each (input/output) training pair, the delta rule determines the direction you need to be adjusted to reduce the error. *Back-propagation* is used for the MLP training (it performs the delta rule). This training algorithm is a supervised learning., e.g. the adjustments to the weights are conducted by back propagating of the error (Haykin, 2001). The use of units with nonlinear activation functions employs the delta rule. Developed by Widrow and Hoff (2001), the delta rule is a version of the least mean square method. For a given input vector x_i, the output vector y_i is compared to the target answer d_i. If the difference is smaller than a required precision, no learning takes place; in the other hand, the weights are adjusted to reduce this difference. the purpose of the learning process, is to minimize the output errors by adjusting the NN synaptic weights w_{ij}.

The generalization is the phase for which NN calculates the corresponding outputs, once it is trained and the NN is ready to receive new inputs (different from training inputs) . Each connection (after training) has an associated weight value that stores the knowledge represented in the experimental problem and considers the input received by each neuron of that NN.

The selection of appropriated NN topology is a complex task, and requires a great effort by an expert, identifying the best parameter set to solve the problems. Generally, the NN topology is usually selected by using empirical or statistical methods that are used to effect a NN and its internal parameters.

MPCA for NN configuration

The methodology developed by Anochi et al. (2013) deals with self-configuration using a new meta-heuristic called the Multiple Particle Collision Algorithm (MPCA) (Luz, 2012) to compute the optimal topology for an MLP. The Particle Collision Algorithm (PCA) starts with a selection of an initial solution, it is modified by a stochastic perturbation leading to the construction of a new solution. The new solution is compared and the new solution can or cannot be accepted. If the new solution is not accepted, the particle can be send to a different location of the search space, giving the algorithm the capability of escaping a local minimum. If a new solution is better than the new solution is absorbed Luz (2012). The implementation of the MPCA algorithm is similar to PCA, but it uses a set with n particles, where a mechanism to share the particles information is necessary.The MPCA objetive function has two terms: one a square difference between NN output and the target data for learning process and a penalty term used to evaluate the complexity for the new network topology at each iteration. The concept of network complexity is associated to the number of neurons and the number of iterations in the training phase. The learning process by MPCA consists to training process to find the connection weights for minimazing the objective function. The initial guest to estimate the optimal set of weights are randomly chosen. The MPCA is a stochastic optimization procedure. Therefore, several realizations are performed. For this application, several realizations are executed with MPCA for NN configurations. The same parameters to set up the MPCA are used to identify the best NN configuration, as to calculate the connection weights.

The main advantage in using an automatic procedure to configure a NN is the ability to define the topology near-optimal NN, without needing the help of experts on the NN approach and/or the application. Such approach avoids this time consuming and tiring process of trial and error to find the optimal neural network topology, see Anochi and Campos Velho (2014).

Local Ensemble Transform Kalman Filter

The analysis is the best estimate of the state of the system based on the optimizing criteria and error estimates. The probabilistic state space formulation and the requirement for updating information when new observations are encountered, are ideally suited to the Bayesian approach like EnKF-based scheme. The EnKF represents the model uncertainty, not by an error covariance matrix, but by an ensemble of point estimates in state space. The ensemble is evolved in time through the full model, which eliminates any need for a linear hypothesis as to the temporal evolution. The ensemble forecasts are used to evaluate the probability distribution.

The EnKF was first proposed by Evensen (1994), it is a sequential method, which means that the model is integrated forward in time and whenever observations are available; these EnKF results are used to reinitialize the model before the integration continues. The algorithm follows the sequential assimilation steps of classical Kalman filter, but it calculates the error covariance matrices as described bellow: each member of the ensemble gets its forecast $\{x^f_{n-1}\}^{(i)} \ : \ i = 1, 2, 3, \cdots, k$, where k is the total members at time t_n, to estimate the state vector $\bar{x}^f$ of the reference model. The ensemble is used to calculate the mean of forecasting $(\bar{x}^f)$:

$$\bar{x}^f \equiv k^{-1} \sum_{i=1}^{k} \{x^f\}^{(i)} \, . \tag{7}$$

Therefore, the model error covariance matrix is:

$$P^f = (k-1)^{-1} \sum_{i=1}^{k} (\{x^f\}^{(i)} - \bar{x}^f)(\{x^f\}^{(i)} - \bar{x}^f)^T. \tag{8}$$

The analysis step determines a state estimate to each ensemble member:

$$\{x^a\}^{(i)} = \{x^f\}^{(i)} + W_K[x^{obs} - H(\{x^f\}^{(i)})] \tag{9}$$

$$W_K = P^f H^T [H P^f H^T + R]^{-1}. \tag{10}$$

The analysis $\{x^a\}^{(i)} \ : \ i = 1, 2, 3, \cdots, k$, (eq. 9) by solving (eq. 10) for W_K to get the optimal weight (e.g. Kalman gain). The matrix H represents the observation operator. The covariance matrix R identifies the observation error. The analysis step also updates the covariance error matrix P^a(eq. 11)

$$P^a = (k-1)^{-1} \sum_{i=1}^{k} (\{x^a\}^{(i)} - \bar{x}^a)(\{x^a\}^{(i)} - \bar{x}^a)^T \tag{11}$$

also an ensemble with the appropriate sample analyses mean

$$\bar{x}^a \equiv k^{-1} \sum_{i=1}^{k} \{x^a\}^{(i)} \, . \tag{12}$$

The LETKF scheme is a model-independent algorithm to estimate the state of a large spatial temporal chaotic system (Ott et al., 2004), it is a EnKF-based scheme. The term ”local” refers to an important feature: it solves the Kalman filter equations locally in model grid space, in which applying a cut-off radius of influence for each observation eliminates spurious correlations. The ensemble transform matrix, composed of the weights of the linear combination, is computed for each local subset of the state vector independently, which allows essentially parallel computations. The local subset depends on the error covariance localization. Typically a local subset of the state vector contains all variables at a grid point. The LETKF scheme first separates a global grid vector into local patch vectors with observations. The basic idea of

LETKF is to perform analysis at each grid point simultaneously using the state variables and all observations in the region centred at given grid point.The local strategy separates groups of neighbouring observations around a central point for a given region of the grid model. Each grid point has a local patch; the number of local vectors is the same as the number of global grid points (Miyoshi and Yamane (2007)).

The code of the LETKF in this experiment is based on the system initially developed by Miyoshi (2005) and has been continuously improved. More information about LETKF can be obtained from Hunt et al. (2007) and Miyoshi and Yamane, 2007).

The FSUGSM model

The two analysis methods (MLP-DA and LETKF) are applied to the Florida State University Global Spectral Model (FSUGSM), that is an atmospheric general circulation model, a computer modeling program: three-dimensional, global, primitive-equation model that generate the entire global circulation of the atmosphere (Smagorinsky, 1983). The dynamical processes are the six primitive equations to forecast atmospheric motion: vorticity, divergence, thermodynamic, continuity, hydrostatic, and moisture, which are expanded in their spectral form. The nonlinear terms are calculated on a Gaussian grid using a transform method. Details, equations and numerical methods can be found in Krishnamurti et al. (1973) and Cocke and Larrow (2000).

The vertical discretization of the FSUGSM uses a finite difference scheme and a semi-implicit leapfrog scheme is used for time integration. The vertical coordinates are defined on sigma ($\sigma = p/p_0$, where p_0 is the pressure surfaces and p is the current level). The horizontal coordinates are latitude and longitude on a Gaussian grid in real space. The spectral model, used in this study, runs with T63 horizontal resolution (approximately $1.875^o \times 1.875^o$) and 27 unevenly spaced vertical levels. A transform technique is applied to calculate the physical processes in real space. The schematic for global model and its physical packages can be seen at Fig. 2 .

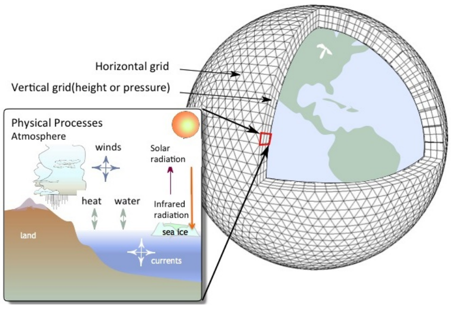


Figure 2 – Schematic for Global Atmospheric Model. Source:Center for Multiscale Modeling of Atmospheric Processes.

The full physical packages include orography, planetary boundary layer, dry adjustment, large-scale precipitation, moist-convection, horizontal diffusion, and radiation processes. The horizontal diffusion term is usually incorporated in a numerical weather prediction model to parameterize the effects of motions on the unresolved scales and to inhibit spectral blocking, that is, the growth of small scales in the dynamic model variables due to the accumulation of energy at high wavenumbers. The presence of any dissipation, physical or computational, can attenuate the amplitude of the short wavelengths very significantly as cited by Zhijin et al. (2000).

The FSU model is global with spectral resolution T63L27 (horizontal truncation of 63 numbers of waves and 27 vertcal levels), corresponding to a regular grid with 192 zonal points (longitude), 96 meridian points (latitude), and 27 vertical sigma levels. The prognostic variables for the model input and output are the absolute temperature (T), surface pressure (p_s), zonal wind component (u), meridional wind component (v), and an additional variable (specific humidity q).

MLP-DA IN ASSIMILATION FOR FSUGSM MODEL

Data assimilation (DA) process generates a model state that is consistent with the observed data, which can be used as an initial condition to run the next model prediction period, is called the DA cycle. The experiment consists of a DA cycle with MLP-DA and a DA cycle of LETKF (Cintra and Cocke, 2013) to obtain the results and comparing their effetive.

The LETKF-FSUGSM is tested with synthetic observations simulating surface pressure and temperature at the model grid point localization. The NN configuration for this experiment is a set of multilayer perceptrons, hereafter referred to as MLP-DA. The MLP-DA was configured with MPCA tools. One strategy used to collect data and to accelerate the processing of the MLP-DA training was to divide the entire globe into four regions: for the Northern Hemisphere, 90^o N and two longitudinal regions of 180^o each; for the Southern Hemisphere, 90^o S and two longitudinal regions of 180^o each. This division provides the same size for each region, and the same number of observations, as illustrated by Fig. 3.

This regional division is applied only for the MLP-DA; the LETKF procedures are not modified.

Firstly, the MLP-DA scheme is developed with a set of sixteen NNs, e.g. four MLPs each region with surface pressure variable (p_s) vectors, and twelve MLPs to four regions with three layers with absolute temperature variable (T) vectors. In this experiment, the MPCA runs with input vectors of surface pressure and temperature simulated observations on some model grid points. The MPCA configuration results are the supervised networks, where the training process needs a target vector for analysis vectors, to each region data to each type of variable, and to each set of trhree layers of upper-air variable. The results of MPCA are:

The MPCA topology

1. Four input nodes, one node for the synthetic observation vector and other for the 6-hours forecast model vector, a node for grid point horizontal coordinate and a node for grid point vertical coordinate.The vectors values represent individual grid points for a single variable with a correspondent observation value;

2. One output node for the analysis vector results. In the training algorithm, the MLP-DA computes the output and compared it with the analysis vector of LETKF cycle results (the target data). The vectors represent the analysis values for one grid point;

3. One hidden layer with *five* neurons;

4. The hyperbolic tangent as the activation function: $\varphi(u_i) = \frac{1-\exp(-au_i)}{1+\exp(-au_i)}$, where $a = 1$, see (Eq. 6);

5. Learning rate η and momentum rate α are the best fitness of MLP-MPCA results. Table1 shows the parameters found by MPCA;

6. Training stops when the error reaches 10^{-5};

7. The MPCA tested the NN configuration in 10 realizations to find the best fitness to sixteen NNs.

Table 1 – Parameters of MPCA topology found to 16 MLPs to MLP-DA.

NETWORK (var/reg/layer)	LEARNING RATE η	MOMENTUM RATE α
ps0101	0.189982	0.535964
ps0201	0.141045	0.678596
ps0301	0.142083	0.419703
ps0401	0.507128	0.808186
tt0101	0.867632	0.242710
tt0102	0.931930	0.857622
tt0103	0.629264	0.767719
tt0201	0.599450	0.880514
tt0202	0.182345	0.393790
tt0203	0.146533	0.129785
tt0301	0.059302	0.124727
tt0302	0.298025	0.666027
tt0303	0.508012	0.899306
tt0401	0.221472	0.129176
tt0402	0.894455	0.388364
tt0403	0.549014	0.778671

The control model fields for this assimilation experiment are obtained from the integration of the model with National Centers for Environmental Prediction (NCEP) reanalysis, i.e. the initial condition to run FSUGSM is the NCEP analysis.The model generates the 6-hours forecast to the entire period of this experiment, considering four times per day (0000, 0600, 1200, 1800 UTC) each January from 2001 through 2003. The data assimilation schemes in this study are based on synthetic observation simulation: where the control model fields are assumed to be known, the observations are simulated by adding Gaussian random noise to the control model; this noise is calibrated according to observational errors. The observational grid is a regularly distributed dense network; it has (45 x 96) points for the p_s and T variables (27 levels). This grid localization is every other latitude/longitude grid point of the FSUGSM native grid of (96 x 192 x 27) (see Fig. 3).

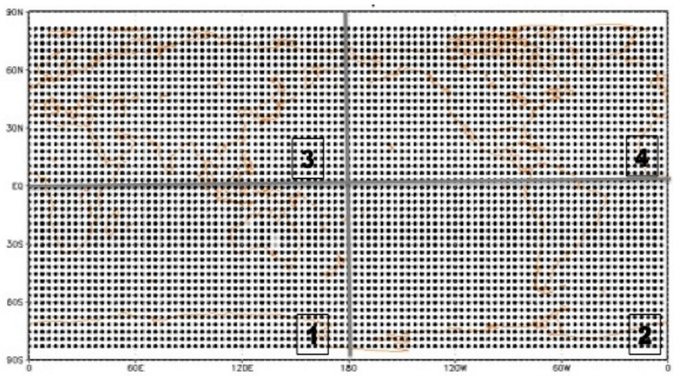


Figure 3 – Observations localizations of Temperature to level 500 hPa, divides in four regions of global area, each is ($90^o \times 180^o$) size. The dot points represent stations.

Training process

Firstly, we run the FSUGSM to generate the control fields and then we perform the observational routine to collect the synthetic observations based on the model fields. The next step is to perform the analysis-forecast cycle for training processes. The first forecast to initiate the analysis cycle is the NCEP analysis field model from 01/01/2001 at 00 UTC and it performs the 6h-forecast and observations to each member for LETKF system. The LETKF system runs with 40 members during the period cited above to simulate observations.

Then, the LETKF scheme is performed, and we obtain the analysis for each member, the forecast mean and analysis mean of all members.We run the FSUGSM model for each analysis member and obtain a member forecast to each member. These forecasts are the first-guess for the next assimilation cycle.

These data are collected for four regions to each chosen grid point, to make NN-MPCA configuration. We use this division strategy to collect the input vectors (observations, mean forecasts, and mean analyses) at chosen grid points by the observations used during LETKF process. The ANN-MPCA process begins after collecting the input vectors for whole period (one month for three years). The ANN-MPCA uses *back-propagation* algorithm that stops the training process using the criteria cited at item 6 above (MPCA topology).

Activation process

The training is performed with combined data from January of 2001 to 2003. MLP-DA is able to perform analyses similar to the LETKF analyses during generalization process in MLP-DA cycles.

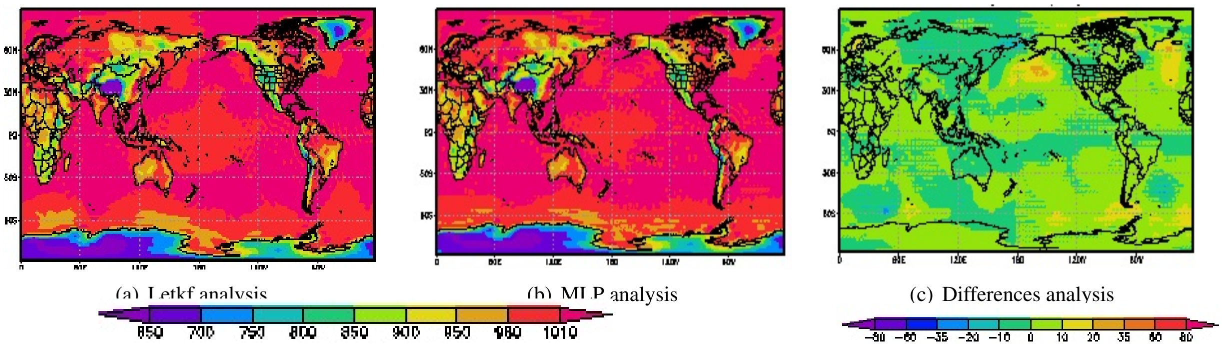
The activation or generalization process is indeed, the data assimilation process. The MLP-DA results a global analysis field. The MLP-DA activation is entering by input values (only 6 hours forecast, observations and coordinatesα) at each grid point once, with no data used in the training process. The input vectors are done at grid model point where is marked with new observation. The procedure is the same for all NNs but one NN for each region and layer to p_s and to T variable, has *different* connection weights. The regional grid points are put in the global domain to make the analysis field after generalization process of the MLP-DA, e.g. the activation of 16 NN results one global analysis.

The MLP-DA data assimilation is performed for one-month cycle, e.g. 124 analysis-forecast cycles. The next assimilation cycle begins as soon as the 6hs-forecast and observations are ready. It starts at 0000 UTC 01 January 2004 NCEP analysis for FSUGSM model producing a 6-hours forecast and observations to performed the initial condition with MLP-DA. The FSUGSM is executed with the former analysis. The process is repeated at each six hours and generates analyses and 6 hours forecasts up through 31 January 2004 1800UTC, to obtain the results.

RESULTS

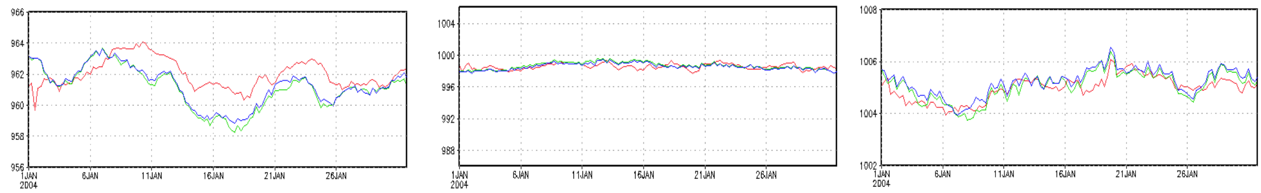
The input and output values of prognostic variable (p_s) are processed on grid model points for time integrations to an intermittent forecasting and analysis cycle.

The results show the comparison of analysis fields, generated by the MLP-DA and the LETKF, and the control model fields. Figure 4 presents the global surface pressure fields in hector-Pascal(hPa) generated from assimilation cycle of 03/Jan/2004 - 06UTC. The differences between analysis from the MLP-DA and LETKF for surface are displayed in Fig. 4(c). The difference for surface pressure are around -10 hPa to 10 hPa. Figure 5 presents the zonal global means from three fixed points at latitude: 30^o North (30N), Equator (EQ) and 30^o South (30S) during the 124 assimilations cycles of January, 2004. The averages are depicted for control model, LETKF analysis, and MLP-DA analysis. The larger disagreement was noted to the MLP-DA at 30N (see Fig. 5(a)).



(a) Letkf analysis (b) MLP analysis (c) Differences analysis

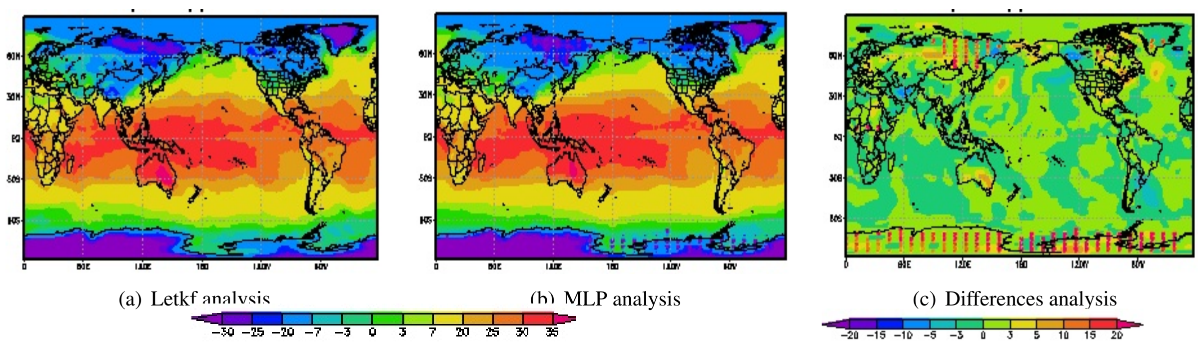
Figure 4 – Surface Pressure (PS) [hPa] Fields 03/01/2004 at 06 UTC. (a) LETKF analysis (b) MLP-DA analysis (c) diferrences between LETKF and MLP-DA analyses.



(a) 30N (b) EQ (c) 30S

Figure 5 – Zonal global mean of surface pressure (hPa) of FSUGSM trajectory during January, 2004 at fixed latitude point: 30^o North (30N), Equator (EQ) and 30^o South (30S). The blue lines are 6-hours LETKF analyses means, the red lines are MLP-DA analyses and green line are the control model.

The global surface temperature (T) in (Celsius degree(C^o)) fields are presented at Fig. 6 generated from aDA cycle of 03/Jan/2004 - 06UTC. The differences between analysis from the MLP-DA and LETKF for temperature are displayed in Fig. 6(c). The differences are around -3 C^o to 3 C^o and we can see some grid observations are around 15 C^o on south pole.



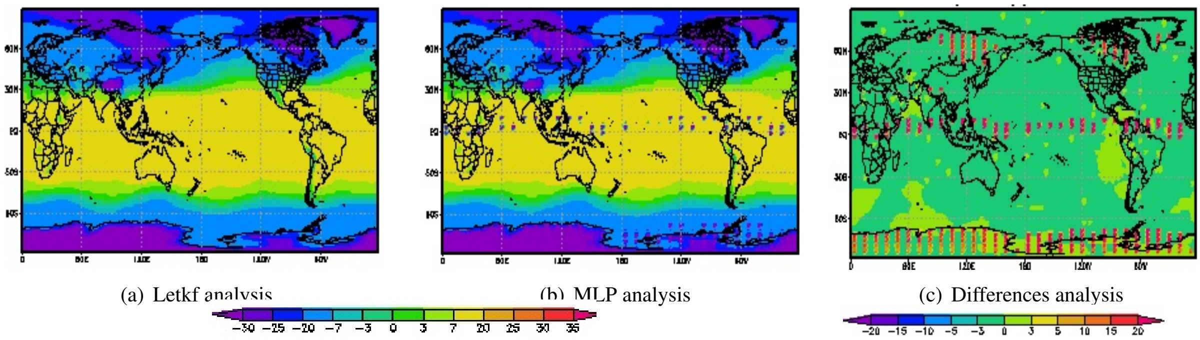
(a) Letkf analysis (b) MLP analysis (c) Differences analysis

Figure 6 – Surface Temperature (C^o) Fields 03/01/2004 at 06 UTC. (a) LETKF analysis (b) MLP-DA analysis (c) diferrences between LETKF and MLP-DA analyses.

Figure 7 presents the global temperature (T) at level 500 hPa fields in Celsius degree(C^o) generated from assimilation cycle of 03/Jan/2004 - 06UTC. The differences between analysis from the MLP-DA and LETKF for temperature are displayed in Fig. 7(c). The differences are around -3 C^o to 5 C^o and some grid observations are around 15C^o or -10C^o on south pole and on equator regions.

CONCLUSION

The MLP-DA data assimilation cycle is composed by the reading of 6-hours forecast of FSUGSM model from latter cycle and reading the set of observations to the cycle time, the division of input vectors, the activation of MLP-DA and the assembly of output vectors to a global analysis field. The comparison in Table 2 is the data assimilation cycles for the same observations points and the same model resolution to the same time simulations. LETKF and MLP-DA executions are performed independently. Considering the total execution time of those 124 cycles simulated with surface pressure



(a) Letkf analysis (b) MLP analysis (c) Differences analysis

Figure 7 – Temperature (C^o) Fields at layer 500 hPa to 03/01/2004 at 06 UTC. (a) LETKF analysis (b) MLP-DA analysis (c) diferrences between LETKF and MLP-DA analyses.

and multilevel temperature variables, the computational performance of the MLP-DA data assimilation, is better than that obtained with the LETKF approach.

Figure 5 shows the differences for the analysis considering the two methods for data assimilation for surface pressure. The LETKF (blue line) presented a better agreement with the reference values (green line) than the MLP-DA (red line). For the North hemisphere (Fig. 5a), there is a larger difference between the MLP-DA and the reference values than the South hemisphere (Fig. 5c), at least for the analysed latitude (30 degree for both hemispheres). Over the equator (Fig. 5b) both schemes for data assimilation has similar results, with LETKF maintaining better output.

These results show that the computational efficiency of the NN for data assimilation to obtain analyses to run the FSUGSM model, for the adopted resolution, is 274 *times* faster and produces analyses of the same quality. To obtain analysis and 6-hours forecasts to January/2004, the NN running time is 59 times faster than LETKF running time cycles based on Table 2.

Table 2 – Total running time of 124 cycles of complete data assimilation (analysis and forecasting) for January/2004

Execution of 124 cycles	MLP-DA (hour:min:sec)	LETKF (hour:min:sec)
Analisys time	00:02:25	11:01:20
Ensemble time	00:00:00	15:50:40
Parallel model time	00:27:20	00:00:00
Total time	00:29:45	26:52:00

Observations data, which are informative to understand weather behaviour, are accumulated every day. The development of sensing technology, increase the acquisition of information automatically. These data come from the global meteorological network and satellites run by countries around the world. However, automatic tracking is challenging under these scenario.

The challenge of numerical weather prediction(NWP) is faced with data volume acquisition and assimilation like: high spectral resolution sounders with data compression, channel selection, reconstructed radiances, with trace gases, aerosols, reactive gases, use of satellite data in cloudy and rainy conditions and so on. The challenges is also, to develop sufficient expertise to exploit a proliferation of new instruments/observing techniques, and the need for developing 'intelligent (dynamical, flow dependent)" data selection to optimise the use os satellite observations and assimilate this information in numerical weather prediction models. The evolution of model resolutions in the horizontal and vertical coordinates, is a challenge too, and the same time, physical parametrizations are improving, and there is a tendency for integrated systems like atmosphere, ocean and land in a variety of applications. These challenges are dependent of computer algorithms and data assimilation techniques that supports these challenges.

REFERENCES

Anochi, J. A. and de Campos Velho, H. F., Foundations of Computational Intelligence (FOCI), 2014 IEEE Symposium on, 128–134,2014.

Anochi, J., Sambatti, S., Luz, E., de Campos Velho, H. F.,2013, "New learning strategy for supervised neural network: MPCA meta-heuristic approach", 1st BRICS Countries & 11th CBIC Brazilian Congress on Computational Intelligence. Location: Recife, Brasil. Porto de Galinhas Beach, September 8th-11th.

Cintra, R. S., 2010, "Assimilação de dados por redes neurais artificiais em um modelo global de circulação atmosférica", D.Sc. dissertation on Applied Computing, National Institute for Space Research, São José dos Campos - Brazil.

Cintra, R. S., Campos Velho, H. F. and Furtado, H.C., 2013, "‘"Neural network for performance improvement in atmospheric prediction systems: data assimilation", 1st BRICS Countries & 11th CBIC Brazilian Congress on Computational Intelligence. Location: Recife, Porto de Galinhas Beach, September 8th-11th, Brazil.

Cintra, R. S. and Cocke, S., 2015, "A local ensemble transform Kalman Filter data assimilation system for the FSU Global Model." Journal of Mechanics Engineering and Automation, No. 5, pp.185-196 doi: 10.17265/2159-5275/2015.03.008.

Cocke, S., and T. E. LaRow,2000, "Seasonal Predictions Using a Regional Spectral Model Embedded within a Coupled Ocean-Atmosphere Model", Mon. Wea. Rev., no. 128, pp. 689-708.

Daley, R, 1991, "Atmospheric Data analysis", Cambridge University Press, New York, USA, 471 pp..

Evensen, G., 1994, "Sequential data assimilation with a nonlinear quasi-geostrophic model using monte carlo methods to forecast error statistics", Journal of Geophysics Research, No. 99, pp. 10143-10162.

Furtado, H. C., 2008, "Redes neurais e diferentes m 'etodos de assimilação de dados em dinâmica não linear", Msc. thesis in Applied Computation program, Brazilian National Institute for Space Research, São José dos Campos, Brazil.

Furtado, H. C. , 2012, "Redes neurais para assimilação de dados em um modelo de circulação oceânica", D.Sc. dissertation on Applied Computing, National Institute for Space Research, São José dos Campos - Brazil.

Ghil M, Malanotte-Rizzoli P. 199t1, "Data assimilation in meteorology and oceanography". Adv. Geophys. 33: 141pp. 266.

Harter, F. P. and. Campos Velho, H. F., 2008, "New approach to applying neural network in nonlinear dynamic model", Applied Mathematical modeling, Vol. 32, No. 12, pp. 2621-2633, DOI:10.1016/j.apm.2007.09.006.

Haykin, S.,2001, Redes neurais princípios prática", Vol. 2, Editora Bookman, Porto Alegre.

Hólm , E. V., 2008, "Lecture notes on assimilation algorithms", Reading, UK, European Centre for Medium-Range Weather Forecasts, 30 pp.

Hsieh, W. and Tang, B., 1998, "Applying neural network models top prediction and data anal- ysis in meteorology and oceanography", Bulletin of the American Meteorological Society, Vol. 79,No. 9, pp. 185-1870.

Hunt, B., Kostelich, E. J. and Szunyogh, I., 2007, "Efficient data assimilation for spatiotem- poral chaos: a local ensemble transform Kalman filter", Physica D, 230, 112-126

Kalman, R. E., 1960, "A new approach to linear filtering and prediction problems", Trans. of the ASME-Journal of Basic Engineering, Vol. 82(Series D), pp. 35-45.

Krishnamurti, T. N., M. Kanamitsu, B. Ceselski, M. B. Mathur, 1973, "Florida State University's Tropical Prediction Model", Tellus , Vol. 25, No. 6, pp. 523-535.

Liaqat, A., Fukuhara, M., and Takeda, T., 2003, "Applying a neural collocation method to an incompletely known dynamical system via weak constraint data assimilation", Monthly Weather Review, Vol. 131, No. 8, pp. 1697-1714.

Lorenz, E.N., 1960, "Generation of available potential energy and the intensity of the general circulation", Tellus, Vol. 12, No. 4, pp. 364.

Luz, E. F. P., 2012, "Meta-heurísticas paralelas na solução de problemas inversos", D.Sc. thesis on Applied Computing Instituto Nacional de Pesquisas Espaciais, São José dos Campos, Brazil.

Miyoshi, T., 2005, "Ensemble Kalman filter experiments with a primitive-equation global model", Ph.D. thesis, University of Maryland, College Park, Maryland, USA, 197 pp.

Miyoshi, T. and. Yamane, S, 2007, "Local ensemble transform Kalman filtering with an AGCM at a T159/L48 resolution", Monthly Weather Review, No. 135, pp. 3841-3861.

Nowosad, A., Neto, A. R.. and Campos Velho, H., 2000, 'Data assimilation in chaotic dynamics using neural networks", International Conference on Nonlinear Dynamics, Chaos, Control and Their Applications in Engineering Sciences, pp. 212.

Ott, E., Hunt, B. R., Szyniogh, I., Zimin, A. V., Kostelich, E. J., Corazza, M., Kalnay, E., Patil, D. J., York, J., 2004, 'A local ensemble Kalman filter for atmospheric data assimilation", Tellus, Vol. A, No. 56, pp. 415-428.

Smagorinsky J., 1983, 'The Beginnings of Numerical Weather Prediction and General Circulation Modeling: Early Recollections", Advances in Geophysics, No. 25, pp. 3-37

Talagrand, O., 2008, "Data Assimilation in Meteorology and Oceanography", Academic Press.

Tang, Y., W. W. Hsieh, B. Tang, and K. Haines, 2001, "A neural network atmospheric model for hybrid coupled modeling", Climate Dynamics, Vol. 17, No.s 5-6, pp. 445-455.

Widrow, B and Hoff,M., 1960, "Adaptive switching circuits", IRE WESCON Conv. Record, Pt.4.94, No. 95, pp. 96-104.

Zhijin, Li , I. Michael Navon and Yanqiu Zhu, 2000, "Performance of 3D-Var with different strategies for the use of adjoint physics with FSU global spectral model", Monthly Weather Review , Vol. 128, No.3, pp. 668-688.

RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.