DOI: 10.14529/cmse140407

INVESTIGATION OF DIFFERENT TOPOLOGIES OF NEURAL NETWORKS FOR DATA ASSIMILATION

F.P. Härter, H.F. Campos Velho

Neural networks have emerged as a novel scheme for a data assimilation process. Neural network techniques are applied for data assimilation in the Lorenz chaotic system. A radial basis function and a multilayer perceptron neural networks are trained employing 1000, 2000, and 4000 examples. Three different observation intervals are used: 0.01, 0.06 and 0.1 s. The performance of the data assimilation technique is investigated for different architectures of these neural networks.

Keywords: data assimilation, Neural Network, Data Assimilation.

Introduction

Data assimilation is a very important process in the numerical weather forecast. It permits the imbedding of observational data in the meteorological model. This data provides a feedback during the generation of the forecast in a real time fashion. However, the process of imbedding the observational data is not straightforward and it has to be done in a very smooth manner in order to minimize the propagation of errors in the forecast model. Usually, the assimilation process can be outlined as a two step iterative process:

Forecast step: $w_n^f = F[w_{n-1}^a]$

Analysis step: $w_n^a = w_n^f + d_n$

where w_n represents model state variable at time step n; $F[.]$ is the mathematical (forecast) model, superscripts f and a denote forecast and analyzed values respectively, and d_n is the innovation of the observational data. Several methods of data assimilation have been developed for air quality problems [1], numerical weather prediction [2], and numerical oceanic simulation [3]. In the case of atmospheric continuous data assimilation there are many deterministic and probabilistic methods. Deterministic approaches include dynamic relaxation, variational methods and Laplace transform, whereas probabilistic approaches include optimal interpolation and Kalman Filtering. In the Kalman filtering, the analysis innovation d_n is computed as a linear function of the misfit between observation (superscript o) and forecast (superscript f):

$$d_n = G_n(w_n^o - H_n w_n^f) \tag{1}$$

where G_n is the weight (gain) matrix, w_n^o is the observed value of w_n and H_n is the observation matrix. The Kalman filter has been tested in strongly nonlinear dynamical systems for assimilation procedure, such as the Lorenz chaotic system. Kalman filtering has the advantage of minimizing the error in the assimilation *plus* propagating this minimized error from one data insertion to the next. However, this process involves a heavy computational load, in particular for large meteorological systems. A strategy to alleviate this load is the use of artificial neural networks (ANN) to emulate the accuracy of the Kalman filtering [4]. Neural networks can be efficiently applied to map two data sets [5]. Several

architectures have been proposed for neural networks. The current work is based on the application of neural networks with *backpropagation* learning for data assimilation.

This paper deals with two neural networks: radial basis function and multilayer perceptron. These ANNs are employed for data assimilation for the Lorenz chaotic system [6]. Three different sizes of training set are used: 1000, 2000, and 4000 examples (patterns). Some numerical experiments are carried out for each training set, considering several time-periods for inserting the observations: 0.01, 0.06, and 0.1 seconds. The quality in the assimilation process is analyzed relating to the number of neurons, and different activation functions in the out-put layer. ANNs with two hidden layers are also studied in a class of experiments.

The next section provides a brief introduction to the neural network architecture used for the data assimilation application, and an outline on Kalman filter is presented too. However, it is not the aim of this paper to present an overview of ANNs. A further section discusses some numerical results. The final section adds some comments and remarks.

1. Non-linear model and assimilation processes

The framework used to perform the numerical experiments for the data assimilation is introduced.

A – The Lorenz Model

The Lorenz system [6] is a hard test for data assimilation, due to the fact it can present a chaotic dynamics. The equations for the Lorenz system are given by

$$\frac{dX}{dt} = -\sigma(X - Y), \tag{2}$$

$$\frac{dY}{dt} = RX - Y - XZ, \tag{3}$$

$$\frac{dZ}{dt} = XY - bZ. \tag{4}$$

This system is integrated using the predictor-corrector method with $\Delta t = 0.001$, using the following initial conditions (the subscript 0 denotes the initial condition): $X_0 = 1.508870$, $Y_0 = -1.531271$, $Z_0 = 25.460910$. The parameters in the system are: $\sigma = 10$, $b = 8/3$, and $R = 28$, so that the system is in the chaotic state.

B – Artificial Neural Networks

ANNs are mathematical models useful for carrying out some learning tasks, such as pattern recognition, function approximation, control, and filtering [5]. Figure 1 displays an outline of an ANN.

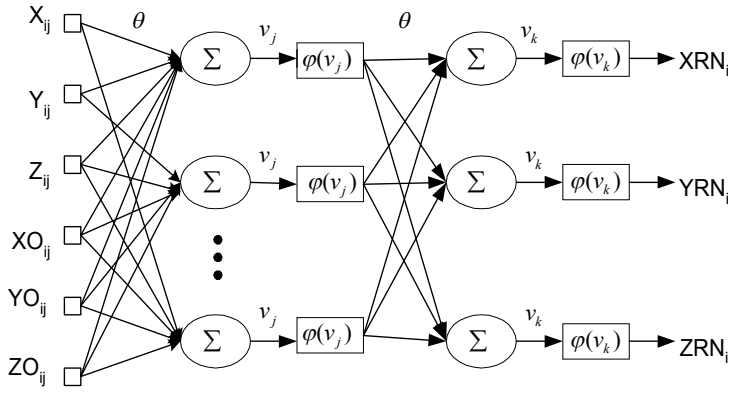

Fig.1. Sketch for neural networks used in this paper

The application of an ANN is done at two phases: learning and activation. The learning phase (also calling *training*) consists to find out the connection synaptic weights and bias associated with each neuron. Two strategies are possible for learning: supervised, and unsupervised. The main difference between supervised and unsupervised learning is that the latter uses only information contained in the input data, whereas the former requires both input and output (desired) data, which allows the calculation of the network error as the difference between the calculated output and the desired vector. In this paper the the supervised backpropagation learning process (Widrow's delta rule) [5] is used.

The activation is the process to obtain an output from an input for a given final architecture of the ANN. The activation function depends on the ANN topology used, for example:

$$\varphi(v_j) = \tanh(\frac{av_j}{2}) \quad (\text{with } a = 1) \tag{5}$$

is employed in the multilayer perceptron, and

$$\varphi(v_j) = \exp\left[-\frac{(v_j - \mu)^2}{2\sigma^2}\right] \quad (\sigma = 1 \text{ and } \mu = 0) \tag{6}$$

is used for radial basis funcions.

Different activation functions can be used for the out-put layer. Functions as given by equations (5) and (6) are tested, as well as linear function: $\varphi(v_j) = v_j$.

Multilayer percentron (MP)

The multilayer perceptron with backpropagation learning, or backpropagation neural network, is a feed-forward network composed of an input layer, an output layer, and a number of hidden layers for extracting high order statistics from the input data. Each of these layers may contain one or more neurons.

Mathematically, a perceptron network simply maps input vectors of real values into output vectors of real values. The connections in figure 1 have associated weights that are adjusted during learning process, thus changing the performance of the network. Neurons in the MP-NN are fully connected.

Radial Basis Functions (RBF)

Girosi and Poggio (1990) [7], based on Kolmogorov's theorem, show that ANNs with only one hidden layer are able to approximate any continuous function. The Girosi and Poggio's proof follows the Idea: a continuous and limited function can be consider as a combination of a linear Gaussians. These Gaussians can be implemented in the hidden layer. The accuracy of the approximation will depend on the number of Gaussian functions, i.e., the number of the neurons in the hidden layer.

ANNs representing functions fitted around a region, whose activation functions, implemented in the neurons of the hidden layer, are Gaussian ones, are examples of the radial basis functions neural network. For this ANN, learning means to find a surface in a multidimensional space, the best fit for the training data, where the agreement is measured in a statistical sense [5].

C – Kalman Filter (KF)

The KF is usefully used in estimation and control problems. Since its first applications on aerospace field [8], this technique has been employed in many applications. Recently, the KF has been applied to meteorology, oceanography and hydrology [2]. A brief description of the Kalman filter will be outlined here. Figure 2 shows an algorithm of the linear KF.

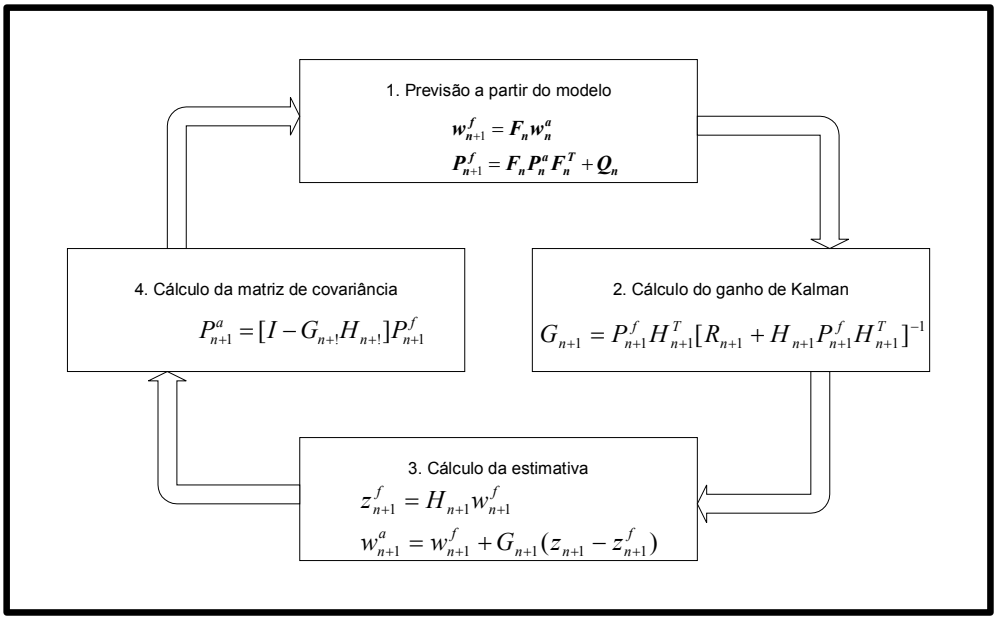

Fig 2. An outline of the Kalman filter algorithm

Let the prediction model be as in equation (7), where the subscript n denotes time-steps.

$$w_{n+1} = F_n w_n + \mu_n \qquad (7)$$

being F_n a mathematical description of the system, and μ_n a stochastic forcing (called dynamic modeling noise), and the observation model

$$z_n = H_n w_n + \upsilon_n \qquad (8)$$

where υ_n is a noise, and H_n represents the observation system. The typical gaussianity, zero-mean and ortogonality hypotheses for the noises are adopted. The term w_{n+1} is estimated through the recursion

$$w_{n+1}^a = (I - G_{n+1}H_{n+1})F_n c_n^a + G_{n+1}z_{n+1} \qquad (9)$$

where w_{n+1}^a is the estimator and G_n is the matrix that minimizes the trace of the prediction error covariance matrix, that is, the sum of the squares of the prediction errors in each component of W_{n+1}

$$J_{n+1} = E\{(w_{n+1}^a - w_{n+1})^T (w_{n+1}^a - w_{n+1})\} \qquad (10)$$

The algorithm of the KF is shown in figure 2, where Q_n is the covariance of μ_n, P_n^f is the covariance of the prediction errors, R_n is the covariance of υ_n, and P_n^a is the covariance of the estimation error. The assimilation is done from the sampled.

$$r(t_n + \Delta t) = r_{n+1} \equiv z_{n+1} - z_{n+1}^f = z_{n+1} - H_n w_{n+1}^f . \qquad (11)$$

2. Results and discussion

As mentioned before, the goal of this paper is to investigate the assimilation system based on ANN with different architectures (MP and RBF). Following this purpose, 396 experiments are performed.

For generating the training sets, the Lorenz system is integrated for 150000 time-steps (0.15 s), sampled at each 30 (0.003 s) producing 5000 examples. The first 4000 examples are applied in the training phase of the ANNs, and the rest of 1000 examples are used for the activation phase. The use of inputs that do not belong to the training set characterizes the *generalization capacity* of the ANN.

Following figure-1, the ANN inputs are normalized matrices $w = w(X, Y, Z)$ of the Lorenz system and $z = z(X_0, Y_0, Z_0)$ is the observation matrix. The desired output is the normalized matrix: $w_a = w_a(X_{FK}, Y_{FK}, Z_{FK})$, resulting from the assimilation with KF. The observations are synthetic ones, adding a Gaussian white noise, with variance 2, to the fields computed from the Lorenz system.

In the back-propagation algorithm, the synaptic weights are initialized according to the Gaussian distribution, and the training patterns are presented in the sequence as generated by the numerical model. ANNs were trained with learning ratio constant and equal to 0.1,

without momentum constant. One difference between MP-ANN and RBF-ANN is in the activation functions of the hidden layer: hyperbolic-tangent – equation (5) – for the former NN, and Gaussian function for the latter one – equation (6).

Figure 3 shows the relevance of the observation system. If there is no assimilation scheme, the disagreement between the computed dynamics (green curve) and true dynamics (observations – blue curve) becomes greater and greater.

Considering the large numbers of experiments, few results are shown. However, comments about our simulations are done.

The ANNs are trained with 1000, 2000, and 4000 examples, with data insertion (assimilation) performed at different time period: 0.01, 0.06, and 0.1 s. The number of neurons in the ANN varies from 3 up to 40, for both ANN. The activation function implemented for the hidden layer of the MP-NN is the hyperbolic-tangent for all experiments, while in the output layer the activation function is linear in the experiments 1 to 11 (C1 set), and hyperbolic-tangent in the experiments 12 to 22 (C2 set). For the RBF-NN the Gaussian function was implemented as activation function in the hidden layer for all experiments, while in the output layer the activation function is linear in the experiments 23 to 33 (C3 set), and Gaussian function in the experiments 34 to 44 (C4 set). Activation functions used here are summarized in the Table I.

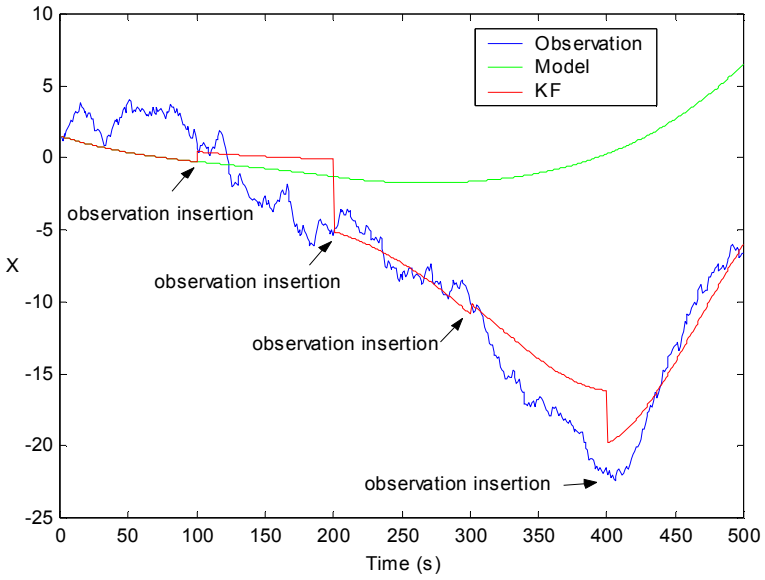

Fig. 3. Importance of the assimilation process

Table 1

Summary of experiments

Experiment	ANN	Output function
EXP1 to EXP11-C1	MP	linear
EXP12 to EXP22-C2	MP	tanh
EXP23 to EXP33-C3	RBF	linear
EXP34 to EXP44-C4	RBF	Gaussian

The quality of an assimilation system can be measured by the quadratic error, after the activation phase. This quantity is computed by the following equation:

$$RMS = \frac{1}{1000}\sum_{i=1}^{1000}(w_a - z)^2 \tag{12}$$

It was observed that when the error reached the value 0.0002 the ANN did not improve the solution. In fact, in many cases the output of the ANN with an error less than 0.0002 degraded the output. Therefore, the error equal 0.002 was defined as the target for training phase. Sometimes, the ANN did not reach this target.

Using 1000 patterns in the training set, with observation sampled at each 0.01 s, both ANNs with one hidden layer produce good results. For some architectures, the assimilation is better than that obtained with KF, whose the error 5.5764. The error for the best result using ANN for the C1 training set is 6.4610 (5 neurons), for o C2 training set is 4.6468 (3 neurons), for C3 training set is 4.5547 (8 neurons), and for C4 training set is 4.7280 (40 neurons). The MP-NN is defined having a linear activation function in the output layer. However, our experiments use hyperbolic-tangent in the output layer, the results are similar or even better when linear function is employed.

Figures 4-5 display the best results for MP-NN with 3 neurons (hyperbolic-tangent as the activation function in the output layer), and RBF-NN with 8 neurons (linear function in the output layer).

The experiments EXP37 (6 neurons) and EXP43 (30 neurons) do not show convergence. Figures 4-5 display the best results for MP-NN with 3 neurons (hyperbolic-tangent as the activation function in the output layer), and RBF-NN with 8 neurons (linear function in the output layer), respectively EXP12 and EXP28 experiments.

Figures 4 and 5 show assimilation results using ANNs (black line) and KF (blue line). Both procedures follow the dynamics of the system. However, one can not see from the figures which ANN produces the best result (smaller **RMS**). Computing **RMS** with equation (12) , the best result is obtained for the MP-NN (3 neurons), with **RMS** a little bit smaller than the best result for the RBF-NN. Experiments also show that ANNs with linear activation function in the output layer present worse results when hyperbolic-tangent and Gaussian functions are used as activation functions in the output layer for the MP-NN and RBF-NN, respectively.

For assimilation with sampled observation 0.06 s the estimative error with is 6.2377, i.e., increasing the time-period of observation the estimative is degraded. However, this conclusion does not apply to the assimilation for some architecture of ANNs. For example, in the C1 training set with sampled observation at 0.06 s, the experiment with 30 neurons presents **RMS** a thin smaller than same experiment sampled observation equal 0.01 s. The same occurs for experiments using C2 training set.

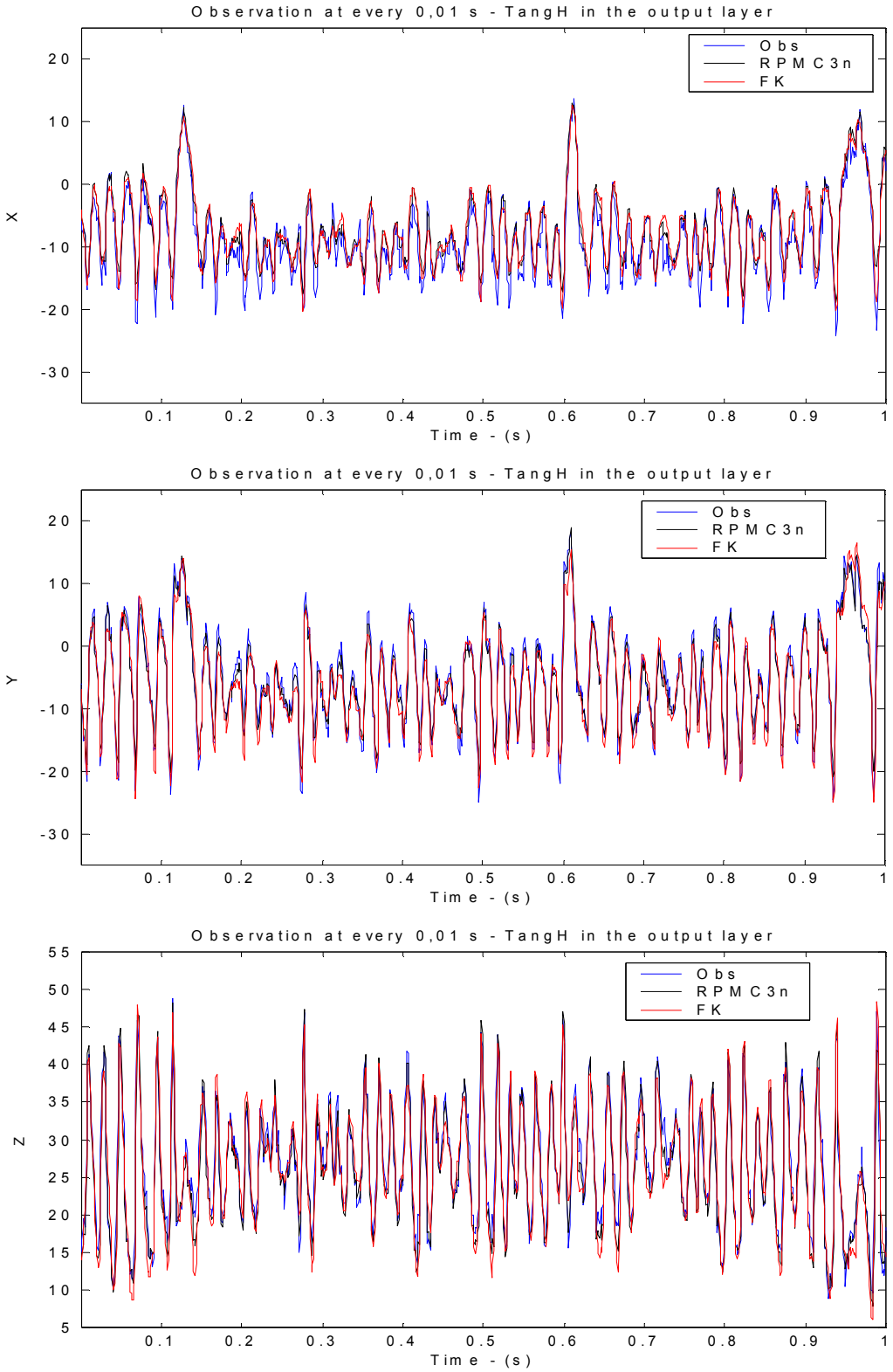

Fig. 4. Assimilation with MP-NN using 3 neurons. Hyperbolic-tangent was used in the output layer

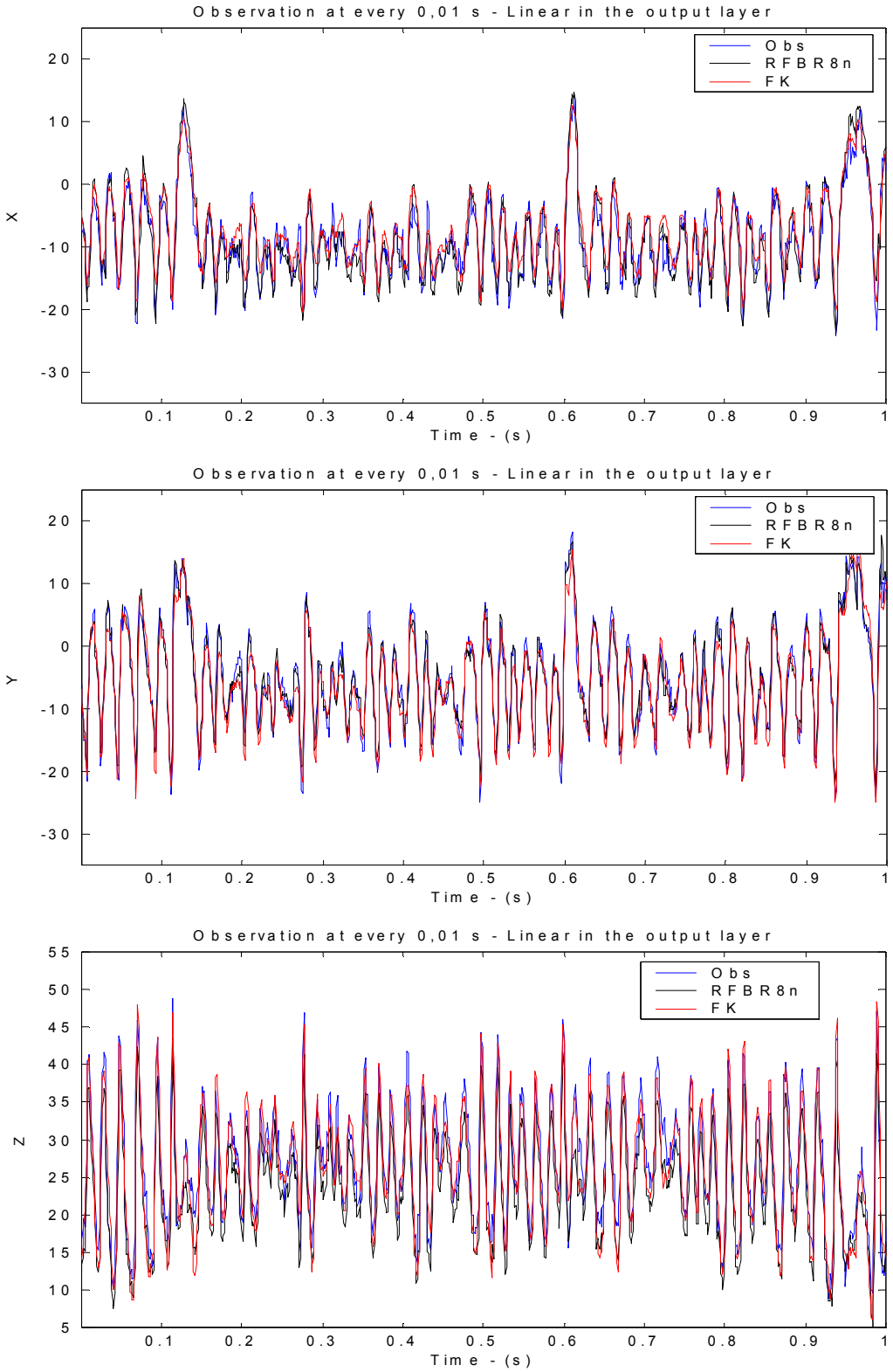

Fig. 5. Assimilation with RBF-NN using 8 neurons. Linear function was used in the output layer

The best results of the MP-NN for sampled observation at 0.06 and 0.01 s were obtained using 3 neurons, with hyperbolic-tangent in the output layer. For RBF-NN, the best

architecture was obtained with linear function in the output layer using 7 neurons and 8 neurons for sampled observation at 0.06 and 0,01 s, respectively.

The analysis of experiments with observations at each 0.1 s, the best architecture was obtained using 7 neurons in the hidden layer, with hyperbolic-tangent in the output layer. For RBF-NN, the best arrangement was gotten using 7 neurons, with linear function in the output layer.

Assimilation observations at each 0.1 s, the error of the estimative by KF increased, related to the experiments in which the assimilation was done at each 0.01 and 0.06 s, showing a **RMS**=9,4537. For the ANNs, the assimilation at each 0.1 s was degraded related to the experiments with observations inserted at 0.01 s, but this tendency is not verified for the most experiments in the C1 and C3 training sets.

It is hard to identify some pattern from the experiments discussed, but as a general conclusion one can say that having more observations (the frequency of the observations sampled) better assimilation can be obtained. Another point is that for MP-NN the use of hyperbolic-tangent in the output layer improves the assimilation. Similar feature is found related to the RBF-NN, using Gaussian function in the output layer. Finally, an obvious point is that architectures with smaller number of neurons are preferred, from computational point of view.

For next, the experiments where ANNs were trained using 2000 examples, with the sampled observations at 0.01 s using 3, up to 40 neurons for both ANNs. The activation function implemented in the hidden layer is the hyperbolic-tangent, for all experiments, while in the output layer the activation function could be linear or hyperbolic-tangent, as shown in Table I. As before, the Gaussian function was used as activation function for the RBF-NN, while linear and Gaussian functions were used in the output layer – see Table I.

The experiments detect *overfiting* using 2000 patterns for training. The estimation presents a large **RMS** value, related to the experiments with 1000 patterns. Estimates with greater period of sampled observation result in a **RMS** greater than those obtained with sampled observation at 0.01 s.

Finally, results obtained using 4000 examples for training are analyzed. The same architecture for MP-NN and RBF-NN used before with 1000 and 2000 examples.

The *overfiting* problem was expected. However, estimations with 4000 examples produce better results than those obtained using 2000 patterns, but the assimilation is worse than 1000 patterns are used. Sampled observations at 0.1 s are indicating better results than those sampled at 0.01 s, for MP-NN.

Results for MP-NN with 2 hidden layers, having sampled observations at 0.01 s, present similar answer to those obtained with only one hidden layer. But, the computational cost for the NN with 2 hidden layer is greater than for a NN with one hidden layer. RBF-NN has one hidden layer, by definition, but some tests using 2 hidden layers were done with this topology, but bad results were obtained.

Conclusion

Artificial neural networks were applied in a assimilation process during the time integration of the Lorenz system in chaotic regime. Tests are done varying the size of the training set, and the time-period of the sampled observations inserted in the integration. The ANNs applied in the assimilation are the MP-NN and RBF-NN with different number of

neurons in the hidden layer(s). The performance of these NNs were also verified related to the use of linear and non-linear activation functions in the output layer.

It is a hard task to find out the optimum architecture for a given NN. However, this is not a constrain for its use, since the problem can be solved with a desired accuracy with simple architecture (few neurons and 1 or 2 layers), implying in a smaller computational cost related to the more complex (bigger) NNs. In few words, having good results, it is not necessary to find the optimum architecture.

The goal of the present study is not to do a formal analysis for each architecture, and the learning strategy employed, instead, the focus here is to show some general tendencies. From these general aspect, we can pointed out that for this application, the use of 1000 examples for training is clearly better than use 2000 or 4000 patterns. The result is better and the computational cost is smaller.

Concerning the time-period for sampled observations, it is important to note that greater time-period do not imply in a worse result, differently of the KF and other traditional schemes. In this work, the best results are obtained inserting observations at each 0.001 s, but greater or less quantity of observational data is characteristic for a given application. In meteorology, observational data are available at 12 and 24 h by the operational meteorological centers, such as NCEP (National Centers for Environmental Prediction) and ECMWF (European Certer for Medium-Range Weather Forecasts). Satellite data have also high interest for data assimilation.

Our experiments suggest that the use of hyperbolic-tangent in the output layer as the activation function of the MP-NN produces better results. The same strategy and conclusion can be applied to the RBF-NN, using the Gassian function in the output layer, since it present smaller sum of the square error.

A future work is to use recurrent neural network for data assimilation. This type of ANN is a system with memory, while the ANNs used in the present paper are memoryless system. Other features that motivate the study of the ANN for data assimilation is that ANNs are essentially parallel algorithms, and they can be implemented in hardware devices.

References

1. Zannetti, P. Air Pollution Modeling, Computational Mechanics Publications, UK, 1990.
2. Daley, R. Atmospheric Data Analysis, Cambridge University Press, Cambridge, 1991.
3. Bennet, A. F. Inverse Methods in Physical Oceanography, Cambridge University Press, 1992.
4. Nowosad, A. G. Data Assimilation Using an Adaptative Kalman Filter and Laplace Transform / A. G. Nowosad, A. Rios Neto, H.F. de Campos Velho // Hybrid Methods in Engineering. – 2000a. –Vol. 2.- P. 291-310.
5. Haykin, S. Neural Networks: A Comprehensive Foundation. Macmillan, New York.
6. Lorenz, E. Deterministic Nonperiodic Flow / E. Lorenz // Journal of Atmospheric Sciences. -1963. – Vol. 20 – P. 130-141.
7. Girosi, F. Networks and the best approximation property/ F. Girosi,T. Poggio // Biological Cibernetic. -1990. –Vol. 45 – P. 169-176. DOI: 10.1007/BF00195855
8. Jazwinski, A. Stochastic Processes and Filtering Theory, Academic Press, New York and London, 1970.

Fabrício Pereira Harter, professor, Faculty of Meteorology, Pelotas Federal University (Pelotas, RS, Brazil), fabricio.harter@ufpel.edu.br.

Haroldo Fraga de Campos Velho, research, Computing and Applied Mathematics, National Institute For Space Research (São José dos Campos, SP, Brazil), haroldo@lac.inpe.br.

Received April 20, 2014.

Bulletin of the South Ural State University
Series "Computational Mathematics and Software Engineering"
2014, vol. 3, no. 4, pp. 96–108

УДК 551.509, 004.94 **DOI: 10.14529/cmse140407**

ИССЛЕДОВАНИЕ РАЗЛИЧНЫХ ВИДОВ ТОПОЛОГИИ НЕЙРОННЫХ СЕТЕЙ ДЛЯ АССИМИЛЯЦИИ ДАННЫХ

Ф.П. Хартер, Г.Ф. де Кампос Вельо

Методы нейронных сетей рассматриваются как альтернатива для существующих схем усвоения наблюдений в геофизические численные модели. Алгоритмы радиальных базисных функций и многослойного перцептрона выбраны для экспериментов по ассимиляции данных в простейшую двумерную гидродинамическую модель, т.н. систему динамического хаоса Лоренца. Обучение обоих типов алгоритмов производилось на выборке из 1000, 2000 и 4000 наблюдений поведения параметров системы с интервалами в 0.01, 0.06 и 0.1 сек, и затем в режиме распознавания произведена сравнительная оценка качества усвоения данных различными архитектурами нейронных сетей.

Ключевые слова: ассимиляция данных, нейронные сети.

Литература

1. Zannetti, P. Air Pollution Modeling, Computational Mechanics Publications, UK, 1990.
2. Daley, R. Atmospheric Data Analysis, Cambridge University Press, Cambridge, 1991.
3. Bennet, A. F. Inverse Methods in Physical Oceanography, Cambridge University Press, 1992.
4. Nowosad, A. G. Data Assimilation Using an Adaptative Kalman Filter and Laplace Transform / A. G. Nowosad, A. Rios Neto, H.F. de Campos Velho // Hybrid Methods in Engineering. – 2000a. –Vol. 2.- P. 291-310.
5. Haykin, S. Neural Networks: A Comprehensive Foundation. Macmillan, New York.
6. Lorenz, E. Deterministic Nonperiodic Flow / E. Lorenz // Journal of Atmospheric Sciences. -1963. – Vol. 20 – P. 130-141.
7. Girosi, F. Networks and the best approximation property/ F. Girosi,T. Poggio // Biological Cibernetic. -1990. –Vol. 45 – P. 169-176.
8. Jazwinski, A. Stochastic Processes and Filtering Theory, Academic Press, New York and London, 1970.

Фабрисио Перейра Хартер, преподаватель метеорологического факультета, Федеральный университет г. Пелотас, Бразилия.

Гарольдо Фрага де Кампос Вельо, исследователь вычислительной и прикладной математики, Национальный институт космических исследований, г. Сан-Жозе-дос-Кампос, Сан Пауло, Бразилия.

Поступила в редакцию 20 апреля 2014 г.