Data Assimilation Using Reconfigurable Computer System

Sabrina B. M. Sambatti[1], Vitor C. Gomes[2], Haroldo F. Campos Velho[1], Andrea Schwertner Charo[3]

[1]Instituto Nacional de Pesquisas Espaciais (INPE) – So Jos dos Campos, SP – Brazil

[2]Instituto de Estudos Avanados/DCTA – So Jos dos Campos, SP – Brazil

[3]Universidade Federal de Santa Maris (UFSM) – Santa Maria, RS – Brazil

{sabrinabms,vconrado}@gmail.com, haroldo@lac.inpe.br, andrea@inf.ufsm.br

Abstract. *An multi-layer perceptron (MLP) neural network (NN) is used to emulate the Kalman filter employed for data assimilation into a linear 1D wave equation. The MLP-NN is implemented on FPGA (Field-Programmable Gate Array). The neural network is automatically configured using the MPCA.*

Resumo. *Uma rede neural (RN) perceptron de múltiplas camadas (PMC) é usada para emular o filtro de Kalman empregado em assimila ção de dados da equa ção da onda linear 1D. A RN-PMC implementada numa FPGA. A rede neural é automaticamente configurada usando o MPCA.*

1. INTRODUCTION

There are modelling errors when a physical processes is represented by a mathematical model. A strategy to mitigate these errors is adding information from the observations. This process is called data assimilation: a method to compute initial condition (*analysis*) combining data from mathematical model and observations [Daley 1993]. Techniques for data assimilation are computationally expensives. Artificial Neural Networks (ANN) can be employed for reducing the computational effort. Previous research has suggested the use of ANN for data assimilation [Harter and Velho 2008, Furtado et al. 2011, Cintra and de Campos Velho 2012]. An automatic configuration approach was adopted to identify the best topology for the supervised NN, where the NN self-configuring is formulated as an optimization problem [Carvalho 2011]. The Multi-Particle Collision Algorithm (MPCA) [Luz et al. 2008] is used to identify the NN configuration [Sambatti et al. 2012]. A trained NN is codified into FPGA. The data assimilation experiment is performed on linear 1D wave equation [Bennett 2002, Furtado et al. 2011].

2. MULTILAYER PERCEPTRON NEURAL NETWORK

MultiLayer Perceptrons (MLP) is one of the most commonly used topologies [Haykin 1994], with at least one hidden layer, and backpropagation algorithm for learning process. The following activation functions can be used here: Gaussian function, logistic, and hyperbolic tangent function.

2.1. Configuration of MLP-ANN by MPCA

The best NN architecture can be determined by considering it as an optimization problem [Carvalho 2011]. Parameters to be evaluated for ANN configuration: number of hidden layers, number of neurons for each hidden layer, type of activation function, learning ratio, and momentum rate. The MPCA is the optimization technique used. The objective function combines two factors: square difference between the target values and the ANN output, and a penalty factor (a measurement for the neural network complexity)[Carvalho 2011]. The penalty function is given by [Carvalho 2011]:

$$penalty = c_1 e^{x^2} + c_2 y + 1 \tag{1}$$

where x is the number of neurons, y corresponds to the number of epochs to convergence, c_1 and c_2 are fitting parameters to find the balance between the factors in measuring complexity. The optimization problem is solved by the MPCA method [Luz et al. 2008], designed to be executed on parallel machines. MPCA is a new version of the standard PCA (Particle Collision Algorithm) [Sacco and De Oliveira 2005]. Differently from the PCA, the MPCA employs multiple particles in a collaborative way.

3. DATA ASSIMILATION

Mathematical models are imprecise approximation for describing a physical phenomena. Observations are inserted to compute a new initial condition (*data assimilation*), mitigating the modelling errors. The Kalman filter is a well established estimation method, and it has been applied to data assimilation [Daley 1993, Kalnay 2003]. Here, the KF will be used as the goal to be emulated by an ANN.

3.1. Data Assimilation by Hardware Devide: FPGA as a Neurocomputer

Cray XD1 is a hybrid system connecting in the same machine CPU and FPGA. There are six interconnected nodes (blades), with two 2.4 GHz AMD Opteron and one Xilinx Virtex II Pro FPGA per node.

The implementation of the MLP-ANN on FPGA, designed for the data assimilation, has different modules. The MAC (Multiplier and Accumulator) unit:product between inputs and weights, adding bias. The next module is the artificial neuron, and it uses a MAC and control structures. The last computational module is a combination of neurons, with the inputs are connected by a unique bus. The neurons can receive data, and the results (outputs) are flowing to the Lookup Table (LUT) unit.

3.2. Linear 1D Wave Equation

The dynamical system used in our tests is a linear, first-order partial differential equation, called here linear 1D wave equation:

$$\frac{\partial \eta}{\partial t} + c\frac{\partial \eta}{\partial x} = F(x,t) \;\; x \in (0, L_x) \; , \;\; \text{and} t > 0 \; , \tag{2}$$

where η is the unknown variable to be estimated by data assimilation, c is the constant phase speed, $F(x,t)$ is the external forcing, with periodic boundary conditions. The initial condition is given by the following equation:

$$\eta(x,0) = \eta_0 \frac{1}{\cosh\{2[(x-v)/\Delta]\}} \qquad 0 \leq x \leq L_x \tag{3}$$

with the same numerical values indicated by [Bennett 2002]. The equation (2) is integrated using finite difference (for space), and Crank-Nicholson method (for time).

4. RESULTS

The linear 1D wave equation was discretized using $N_x = 128$ grid points for the space mesh, with time integration up to 200 time-steps. For the Kalman filter implementation, the covariances adopted by [Furtado et al. 2011] are assumed here. The observations were synthetically generated – with 5% for level of noise, and the assimilation cycle is carried out every 10 time-steps.

The topology for the neural network trained to emulate the Kalman filter was obtained with MPCA, using: 6 particles (meaning six processors), number of iterations = 20. The best architecture found by MPCA: one hidden layer, 23 neurons, learning ratio = 0.4, momentum = 0.6, and tanh(x) as the activation function. Two different topologies of ANN were implemented on FPGA: the MLP-NN designed by experts (Furtado, Campos Velho and Macau, 2011), and the MLP-NN2: the best topology defined by MPCA. The MLP-ANN was implemented for the hybrid system Cray XD1.

To evaluate the results of FPGA-neurocomputer implementations, a version was implemented on software (CPU: Central Processing Unit). The results are shown in Figure 1, corresponding on software and FPGA implementations. The values showed in Table 1 correspond to average square error, and variance.

Table 1. Results

	ANN-2	ANN-3
Average square error	$1.68 * 10^{-5}$	$5.18 * 10^{-5}$
Variance	$1.69 * 10^{-9}$	$1.79 * 10^{-8}$

5. CONCLUSION

Artificial neural networks can be designed as a method for data assimilation. Here, the MLP-ANN was applied to emulate the Kalman filter to the wave 1D dynamical system.

The implementation on FPGA works well, where the fixed point arithmetic was adopted for avoiding memory constraints. In the FPGA implementation, the activation function is not codified as a mathematical function, a lookup table approach was employed. The strategy for the automatic configuration of the MLP-ANN using the MPCA was effective. The computed ANN topology produced better results than a configuration defined by an expert.

References

Bennett, A. F. (2002). *Inverse Modeling of The Ocean and Atmosphere*. Cambridge University Press.

Carvalho, A. R. (2011). *Uso de redes neurais para recuperação do perfil de concentração de gases traço atmosféricos a partir de dados de satélites*. Tese de doutorado, Instituto Nacional de Pesquisas Espaciais (INPE), São José dos Campos.

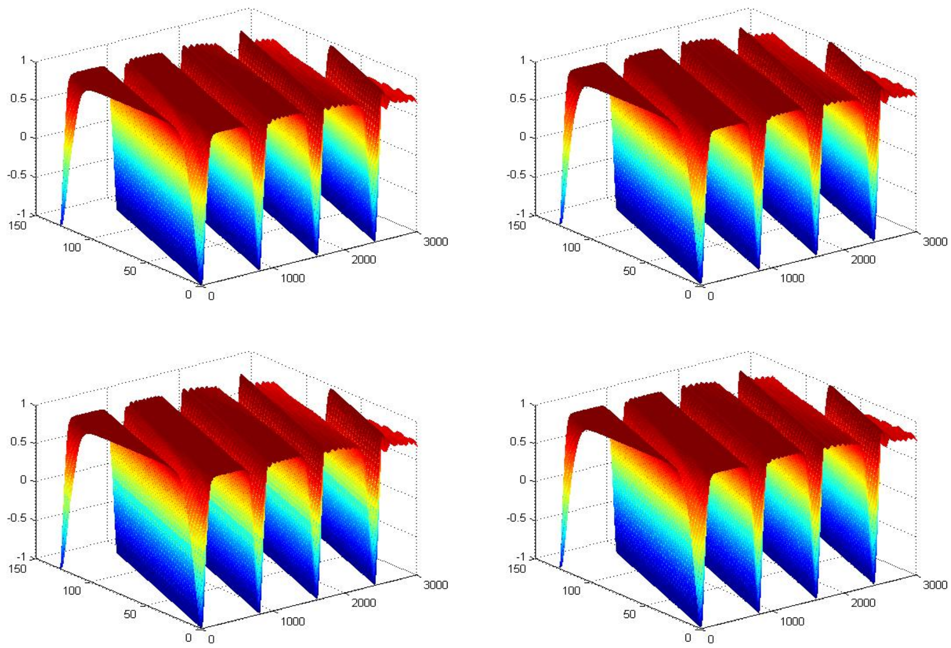

Figure 1. Projection of the linear 1D wave model at coordinates x (position), y (time), z (wave spread). Left: empirical ANN (upper: software,bottom: FPGA). Right: MPCA-ANN2 (upper: software,bottom: FPGA).

Cintra, R. and de Campos Velho, H. F. (2012). Global data assimilation using artificial neural networks in seedy mode. In *1st International Symposium Uncertainty Quantification and Stochastic Modeling*, pages 648 – 654, Maresias.

Daley, R. (1993). *Atmospheric data analysis*, volume 2. Cambridge university press.

Furtado, H., de Campos Velho, H., and Macau, E. (2011). Assimilação de dados com redes neurais artificiais em equações diferenciais. *Anais do DINCON*, pages 595 – 598.

Harter, F. P. and Velho, H. F. C. (2008). New approach to applying neural network in nonlinear dynamic model. *Applied Mathematical Modeling*, 32(12):2621 – 2633.

Haykin, S. (1994). *Neural networks: a comprehensive foundation*. Prentice Hall Inc.

Kalnay, E. (2003). *Atmospheric modeling, data assimilation, and predictability*. Cambridge university press.

Luz, E. F. P., Becceneri, J. C., and de Campos Velho, H. F. (2008). A new multi-particle collision algorithm for optimization in a high performance environment. *Journal of Computational Interdisciplinary Sciences*, 1(1):3 – 10.

Sacco, W. F. and De Oliveira, C. R. (2005). A new stochastic optimization algorithm based on a particle collision metaheuristic. *Proceedings of 6th WCSMO*.

Sambatti, S. B. M., Anochi, J. A., da Luz, E. F. P., Carvalho, A. R., Shiguemori, E., and de Campos Velho, H. F. (2012). Automatic configuration for neural network applied to atmospheric temperature profile identification. *International Conference on Engineering Optimization*.