A TWO STEP INVERSE PROBLEM FOR VERTICAL TEMPERATURE PROFILE RETRIEVAL IN CLOUDY ATMOSPHERE USING ARTIFICIAL NEURAL NETWORKS

Patrícia Oliva Soares
Leopoldo Américo Miguez de Mello Research and Development Center (Cenpes), Rio de Janeiro (RJ), Brazil
oliva@petrobras.com.br

Antônio José da Silva Neto
Politechnique Institute (IPRJ), Rio de Janeiro State University (UERJ), Nova Friburgo (RJ), Brazil
ajsneto@iprj.uerj.br

Haroldo Fraga de Campos Velho
Associated Laboratory for Computing and Applied Mathematics (LAC), National Institute for Space Research (INPE), São José dos Campos (SP), Brazil
haroldo@lac.inpe.br

Francisco José da Cunha Pires Soeiro
Department of Mechanical Engineering, College of Engineering, Rio de Janeiro State University (UERJ), Rio de Janeiro (RJ), Brazil
soeiro@uerj.br

Abstract. A methodology for retrieving vertical temperature profiles in the atmosphere with clouds from radiance measurements made by satellite, using artificial neural networks, is presented. Vertical temperature profiles are important initial conditions for numerical weather prediction models, and are usually obtained from measurements of radiance using infrared channels. Though, when these measurements are performed in a cloudy atmosphere, it is not possible to retrieve the temperature profile with current techniques. It represents a considerable loss of information, since on average 20% of the pixels of the images have clouds. The methodology proposed here consists of a two step inverse problem. The first step is an inverse problem of boundary condition estimation, where the radiance reaching the cloud basis is determined from radiance measured by satellite. The second step consists in determining the vertical temperature profile from the boundary condition estimated in the first step. The results obtained for four test cases show that the methodology produces satisfactory results and has great potential for use, allowing to incorporate information from a wider area of the planet and thus to improve numerical weather prediction models.

Keywords: inverse problems, radiative transfer, artificial neural networks, remote sensing, clouds

1. INTRODUCTION

The numerical weather prediction is one remarkable scientific conquer for the twentieth century. Firstly, the weather prediction was identified as an initial value problem at 1904 (known as Vilhelm Bjerknes' theorem). However, the motion equation for fluids is a very complicated problem. Indeed, there is no mathematical theory for solving the Navier Stokes by analytical methods. In 1922, Lewis Fry Richardson has published a book proposing the weather predicting by solving the complete Navier-Stokes using finite differences (Richardson, 1922). The scheme proposed by Richardson failed; Lynch (2006) did an interesting report for the process, and an overview on the modern weather forecasting can be found on Haltiner and Willias (1980).

In order to have a better forecasting, the computation of a good initial condition is a mandatory issue. This imposes the necessity of using observed data from the regions for which the forecasting will be performed. In this sense, meteorological satellites can provide information from regions with difficult access, as tropical rain forests, deserts, poles of the planet, and over the ocean, overcoming the lack of information provided by terrestrial measurement stations. The mathematical equation relating the satellite radiances and atmospheric profiles (RTE: Radiative Transfer Equation) describes the physical conception behind the identification process for estimating temperature profile, moisture, and gas concentration. The forward problem is to compute the radiances, and the inverse problem is to retrieve the properties from the measured radiances.

Several approaches to retrieve the atmospheric temperature profile from satellite data have been developed. In the infrared band from the solar spectrum, the temperature profile can be calculated for clear atmosphere. However, more than 20% of images contain clouds. Under this situation, micro-wave band can be used, with worse precision, to compute the temperature profile, but for a few classes of clouds (cirrus and thin stratus, for example). For deeper clouds (stratus-cumulus and cumulus-nimbus), nor micro-wave can provide useful information. Therefore, there is no information for a considerable part of the Earth.

The goal of the present paper is to propose and evaluate an algorithm to identify atmospheric temperature profile under all classes of clouds. The procedure can also be applied to other atmospheric properties, as humidity and gas concentration. The idea is to combine two inversion procedures in the same framework: reconstruction of cloud bottom boundary condition for RTE (Campos Velho, *et al.*, 2002), and then retrieving the temperature profile (Shiguemori, *et al.*, 2006), as shown schematically at Fig. 1. For step-1, the RTE is a complete Boltzmann equation, which includes the scattering process. In step-2, the RTE is expressed by the Schwarzchild's equation, and the scattering phenomenon can be neglected.
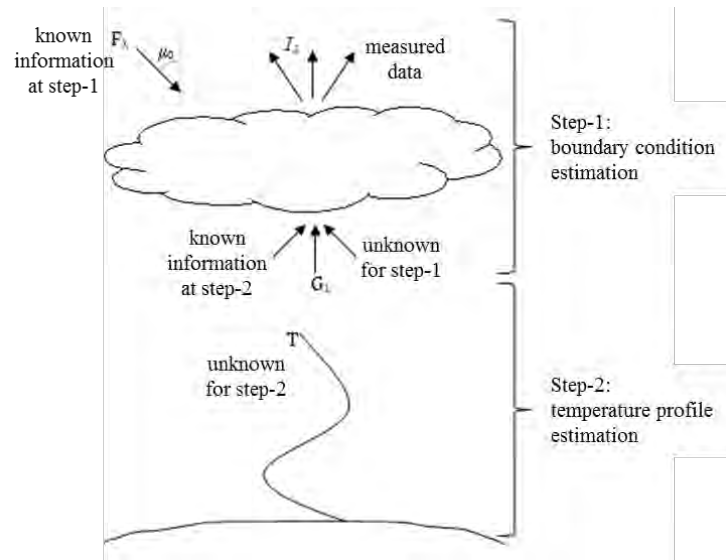

Figure 1. Scheme of the proposed methodology

For both steps, two different multi-layer perceptron (MLP) artificial neural networks (ANN) are used as inversion operators. The best configuration for the ANN is obtained automatically, by using self-configured ANN strategy (Carvalho, *et al.*, 2010), where an objective function is minimized in order to obtain an ANN with few neurons and fast convergence. Multi-particle collision algorithm (Luz, *et al.*, 2008) is the metaheuristic used for solving the optimization problem, and it has shown good results to find out the optimized ANN.

Noisy synthetic satellite data were used to test the inverse methodology, and preliminary results confirm the good performance of the proposed technique.

2. SOLVING THE RADIATIVE TRANSFER EQUATION

The radiative transfer equation models the light beam interactions into a medium. Light intensity is given by a directional quantity, the radiance I, which measures the rate of energy being transported at a given point and for a given direction. Considering a horizontal plane, this direction is defined by a polar angle θ (relative to the normal of the plane) and an azimuthal angle, ϕ. Once within the medium, light can be absorbed, scattered or transmitted, according to the absorption and scattering coefficients, κ_λ and σ_λ, and to a scattering phase function, p_λ, describing how light is scattered in any direction. An extinction coefficient β_λ is defined as $\beta_\lambda = \kappa_\lambda + \sigma_\lambda$.

Assuming plane parallel atmosphere, considered a good approximation for planetary atmospheres (Lenoble, 1993; Özişik, 1973), where variations of all quantities at vertical direction are too much fast than that a horizontal directions, the RTE is given by

$$\frac{\mu}{\beta_\lambda}\frac{\partial I_\lambda(z;\mu,\phi)}{\partial z} + I_\lambda(z;\mu,\phi) = S_\lambda(z;\mu,\phi), \tag{1}$$

where

$$S_\lambda(z;\mu,\phi) \equiv (1-\varpi_\lambda)B_\lambda(T) + \frac{\varpi_\lambda}{4\pi}\int_{\phi=0}^{2\pi}\int_{\mu=-1}^{1} I_\lambda(z;\mu',\phi')p_\lambda(\mu',\phi';\mu,\phi)d\mu'd\phi', \tag{2}$$

being $\mu \in [-1,1]$ the cosine of the incident polar angle θ, $\varpi_\lambda \equiv \sigma_\lambda(s)/\beta_\lambda(s)$ the single scattering albedo and $B_\lambda(T)$ is the Planck function (Planck, 1901),

$$B_\lambda(T) = \frac{2hc^2}{\lambda^5 \left(e^{hc/k_B\lambda T} - 1\right)}, \tag{3}$$

where h is the Planck constant, c is the speed of light at vacuum, k_B the Boltzmann constant and T the absolute temperature.

2.1 Boltzmann equation

For step-1, the medium is a cloud, where there is no radiation source term. Under these hypothesis, and considering the definition of the optical depth (with: $d\tau \equiv \beta_\lambda dz$), Eqs. (1) and (2) can be rewritten as

$$\mu \frac{dI_\lambda(\tau;\mu,\phi)}{d\tau} = -I_\lambda(\tau;\mu,\phi) + \frac{\varpi_0}{4\pi} \int_{\phi=0}^{2\pi} \int_{\mu=-1}^{1} I_\lambda(\tau;\mu',\phi') p_\lambda(\mu',\phi';\mu,\phi) d\mu' d\phi', \tag{4}$$

known as Boltzmann equation, subjected to boundary conditions

$$I_\lambda(0;\mu,\phi) = F\delta(\mu - \mu_0)\delta(\phi - \phi_0) \tag{4a}$$
$$I_\lambda(\tau_0;-\mu,\phi) = G\delta(\mu - \mu_1)\delta(\phi - \phi_1) \tag{4b}$$

where (μ_0, ϕ_0) and (μ_1, ϕ_1) represent the polar and azimuthal directions of incident beams, F and G, at $\tau = 0$ (top of the cloud) and $\tau = \tau_0$ (bottom of the cloud).

To simplify the solution and to use some previous development done by Souto et al. (2006), the problem defined by Eq. (4) is split in two ones, assuming that the incident beam at one boundary does not interact with the incident beam at the other boundary. Therefore,

$$\mu \frac{dI_\lambda(\tau;\mu,\phi)}{d\tau} = -I_\lambda(\tau;\mu,\phi) + \frac{\varpi_0}{4\pi} \int_0^{2\pi} \int_{-1}^{1} I_\lambda(\tau;\mu',\phi') p(\mu,\phi;\mu',\phi') d\mu' d\phi' \tag{5}$$
$$I_\lambda(0;\mu,\phi) = F\delta(\mu - \mu_0)\delta(\phi - \phi_0) \tag{5a}$$
$$I_\lambda(\tau_0;-\mu,\phi) = 0 \tag{5b}$$

and

$$\mu \frac{dI_\lambda(\tau;\mu,\phi)}{d\tau} = -I_\lambda(\tau;\mu,\phi) + \frac{\varpi_0}{4\pi} \int_0^{2\pi} \int_{-1}^{1} I_\lambda(\tau;\mu',\phi') p(\mu,\phi;\mu',\phi') d\mu' d\phi' \tag{6}$$
$$I_\lambda(0;\mu,\phi) = 0 \tag{6a}$$
$$I_\lambda(\tau_0;-\mu,\phi) = G\delta(\mu - \mu_1)\delta(\phi - \phi_1) \tag{6b}$$

and the final solution is the sum of the solutions of both problems. Both problems are solved using the same technique, and only the solution for Eq. (5) will be shown here.

Let us assume that the electromagnetic radiation field consists of two components: one called unscattered component (I_u), and the scattered component (I_s). Thus, the total radiation can be written as the sum of these two components,

$$I(\tau;\mu,\phi) = I_u(\tau;\mu,\phi) + I_s(\tau;\mu,\phi). \tag{7}$$

The formulation for the unscattered component is obtained eliminating the scattering term (ϖ_0) in Eq. 5, and its explicit solution is given by

$$I_u(\tau;\mu,\phi) = F\delta(\mu - \mu_0)\delta(\phi - \phi_0)e^{-\tau/\mu_0}, \tag{8}$$

for $\tau \in [0, \tau_0]$, $\mu \in (0,1]$ and $\phi \in [0, 2\pi]$. Using Eq. (7) at Eq. (5), and considering Eq. (8), one obtains

$$\mu \frac{dI_s(\tau;\mu,\phi)}{d\tau} = -I_s(\tau;\mu,\phi) + \frac{\varpi_0}{4\pi} \int_0^{2\pi} \int_{-1}^{1} I_s(\tau;\mu',\phi') p(\mu,\phi;\mu',\phi') d\mu' d\phi'$$
$$+ \frac{\varpi_0}{4\pi} p(\mu,\phi;\mu_0,\phi_0) Fe^{-\tau/\mu_0} \tag{9}$$

$$I_s(0; \mu, \phi) = 0 \qquad (9a)$$
$$I_s(\tau_0; -\mu, \phi) = 0 \qquad (9b)$$

The above is solved by the LTSN method (Barichello and Vilhena, 1993). This method consists on the application of the Laplace transform on the radiative transfer discrete ordinates equations, producing and algebraic equation on s, and then the application of inverse Laplace transform to obtain the desired radiation field.

For obtaining the radiative transfer discrete ordinates equations from Eq. (9), first the anisotropic scattering phase function p_λ is represented by an expansion in Legendre polynomials P_l in terms of the cosine of the scattering angle θ. After that, the polar angle domain (μ) is discretized, and the integral is approximated by a Gauss-Legendre quadrature:

$$\mu_j \frac{dI^m(\tau, \mu_j)}{d\tau} = -I^m(\tau, \mu_j) + \frac{\varpi_0}{2} \sum_{l=m}^{L} \omega_l^m P_l^m(\mu_j) \sum_{i=1}^{N} \eta_i P_l^m(\mu_i) I^m(\tau, \mu_i) + S^m(\tau, \mu_j) \qquad (10)$$

for $j = 1, 2, \ldots, N$, where η_i are the weights of the quadrature, the boundary conditions are described as

$$I^m(0, \mu_j) = 0, \quad j = 1, 2, \ldots, n \qquad (10a)$$
$$I^m(\tau_0, \mu_j) = 0, \quad j = n+1, n+2, \ldots, N \qquad (10b)$$

and S^m is the source scattering function, given by

$$S^m(\tau, \mu_j) = \frac{\varpi_0}{2\pi} \left[\sum_{l=m}^{L} \omega_l^m P_l^m(\mu_j) P_l^m(\mu_0) \right] Fe^{-\tau/\mu_0}. \qquad (11)$$

Applying Laplace transform on Eq. (10), the following system of equations on s is obtained (Barichello and Vilhena, 1993),

$$s\bar{I}^m(s) + \frac{1}{\mu_j} \bar{I}^m(s) - \frac{\varpi_0}{2\mu_j} \sum_{l=m}^{L} \beta_l^m P_l^m(\mu_j) \sum_{i=1}^{N} \eta_i P_l^m(\mu_i) \bar{I}^m(s) = I_j^m(0) + \frac{1}{\mu_j} \bar{S}^m(s), \qquad (12)$$

where $\bar{I}^m(s) = \int_0^\infty I^m(\tau) e^{-s\tau} d\tau$. The matrix form of Eq. (12) becomes

$$\bar{M}_N^m(s) \bar{I}^m(s) = I^m(0) + \bar{S}^m(s), \qquad (13)$$

where the matrix $\bar{M}_N^m(s)$, called the LTSN matrix, is given by $\bar{M}_N^m(s) = sI + A^m$, where I is the identity matrix and the A matrix is given by

$$a^m(i,j) = \begin{cases} \dfrac{1}{\mu_j} - \dfrac{\varpi_0}{2\mu_j} \displaystyle\sum_{l=m}^{L} \beta_l^m P_l^m(\mu_j) \eta_j P_l^m(\mu_j), & if \ i = j \\[4mm] -\dfrac{\varpi_0}{2\mu_j} \displaystyle\sum_{l=m}^{L} \beta_l^m P_l^m(\mu_j) \eta_i P_l^m(\mu_i), & if \ i \neq j \end{cases}, \qquad (14)$$

while vectors $\bar{I}^m(s)$, $I^m(0)$ and $\bar{S}^m(s)$ are defined as

$$\bar{I}^m(s) = [\bar{I}_1^m(s) \ \bar{I}_2^m(s) \ \ldots \ \bar{I}_N^m(s)] \qquad (15)$$
$$I^m(0) = [I_1^m(0) \ I_2^m(0) \ \ldots \ I_N^m(0)] \qquad (16)$$
$$\bar{S}^m(s) = \left[\frac{\bar{S}_1^m(s)}{\mu_1} \ \frac{S_2^m(s)}{\mu_2} \ \ldots \ \frac{S_N^m(s)}{\mu_N} \right]. \qquad (17)$$

In order to solve Eq. (13), it must be multiplied by the inverse matrix of $\bar{M}_N^m(s)$, represented by $\bar{B}^m(s)$, resulting in $\bar{I}^m(s) = \bar{B}^m(s) I^m(0) + \bar{B}^m(s) \bar{S}^m(s)$. The inverse Laplace transform is applied to solve this equation, obtaining the electromagnetic radiation field,

$$I^m(\tau) = B^m(\tau) I^m(0) + H^m(\tau), \qquad (18)$$

where $B^m(\tau) = \mathcal{L}^{-1}[\bar{B}^m(s)]$ and $H^m(\tau) = B^m(\tau) * S^m(\tau)$, and the convolution operation is denoted by $*$.

2.2 Schwarzchild equation

For step-2, the medium is the atmosphere between the ground and the bottom of the cloud, that can be considered a non-scattering medium, under local thermodynamic equilibrium. From these considerations, the source function represented by Eq. (2) is simplified to Planck function. Equations (1) and (2) can be rewritten as (Liou, 2002)

$$\mu \frac{dI_\lambda(\tau; \mu, \phi)}{d\tau} + I_\lambda(\tau; \mu, \phi) = B_\lambda(T), \tag{19}$$

known as Schwarzchild equation, widely used on remote sensing of atmospheric profiles.

The atmospheric radiation is assumed independent on azimuthal angle. The amount of radiation reaching a certain level τ of the atmosphere is calculated multiplying Eq. (19) by $e^{-\tau/\mu}$ and integrating between τ and τ_1, where $\tau = \tau_1$ refers to the ground surface. For remote sensing applications, it is assumed $\mu = cos\,\theta \approx 1$, and writing the result in terms of pressure coordinates

$$I(\nu_i) = B[\nu_i, T(p_s)]\Im(\nu_i, p_s) + \int_{p_s}^{p_0} B[\nu_i, T(p)] \frac{\partial \Im(\nu_i, p)}{\partial p}\, dp, \tag{20}$$

where p is the atmospheric pressure, p_0 the pressure at the top of the atmosphere, p_s the pressure at the surface and $\Im(z) = e^{-\tau}$ the monochromatic transmittance. The quantity $\partial\Im(\nu_i, p)/\partial p$ is known as weighting function, and when multiplied by Planck function provides a measure of the contribution of each atmospheric layer to the radiation that reaches a specified pressure level dp. Eq. (20) is solved using finite difference method,

$$I_i = B_{i,s}(T_s)\Im_{i,s} + \sum_{j=1}^{N_p} \left(\frac{B_{i,j} + B_{i,j-1}}{2} \right) [\Im_{i,j} - \Im_{i,j-1}], \tag{21}$$

for $i = 1, 2, \ldots, N_\lambda$, where N_λ is the number of spectral channels and N_p the number of atmospheric layers being considered.

3. SELF-CONFIGURED ARTIFICIAL NEURAL NETWORK

Artificial neural networks (ANN) are massively parallel and distributed structures inspired by the operation of biological neural network of human brain. They resemble the brain in two main aspects: they acquire knowledge from the environment through a learning process and store this knowledge on connection or synaptic weights. Moreover, they also have the ability of generalize, meaning they can produce appropriate outputs from inputs that have not been used during the learning process. The abilities of learning and generalization make ANN able to solve complex problems, especially highly nonlinear problems (Haykin, 2001).

Artificial neural networks are made of arrangements of processing elements, called neurons. As represented on Fig. 2, the artificial neuron model basically consists of a linear combiner, followed by an activation function, given by

$$y_k = \varphi \left(\sum_{j=1}^{N} w_{kj} x_j + b_k \right), \tag{22}$$

where x_j is the input vector, w_{kj} are the connection weights, b_k is the bias, y_k is the output of the kth neuron and $\varphi(\cdot)$ is the activation function. used to limit the output of the neuron to a finite value.
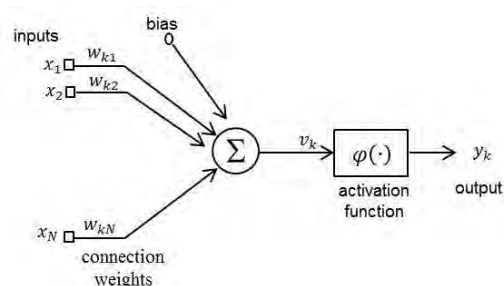

Figure 2. Single neuron model

P. Oliva Soares, A.J. Silva Neto, H.F. Campos Velho, F.J.C.P. Soeiro
A Two Step Inverse Problem for Vertical Temperature Profile Retrieval in Cloudy Atmosphere Using Artificial Neural Networks

Three activation functions are considered in this work. The logistic function, with image between 0 and 1, is defined by

$$\varphi(\vartheta) = \frac{1}{1 + \exp(-a\vartheta)}, \tag{23}$$

where a is a slope parameter, considered here equal to one (Fig. 3a). The hyperbolic tangent function, with image between -1 and 1, is defined by $\varphi(\vartheta) = \tanh(\vartheta)$ and represented on Fig. 3b. Finally, we also consider the Gaussian function, written as $\varphi(\vartheta) = \exp{-(\vartheta^2)}$, shown on Fig. 3c.
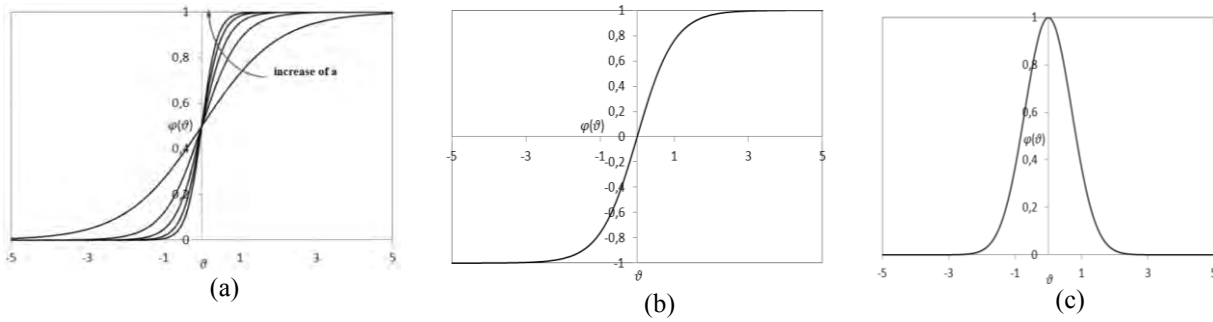

Figure 3. Activation function. (a) Logistic. (b) Hyperbolic tangent. (c) Gaussian

There are several different architectures for ANN. The Multilayer Perceptron (MLP), with backpropagation learning process, is our choice. Multilayer perceptrons are feedforward networks composed of an input layer, one output layer and a number of hidden layers, whose aim to extract high order statistics from the input data, as represented in Fig. 4. The number of neurons at input and output layers are defined by the problem being solved.
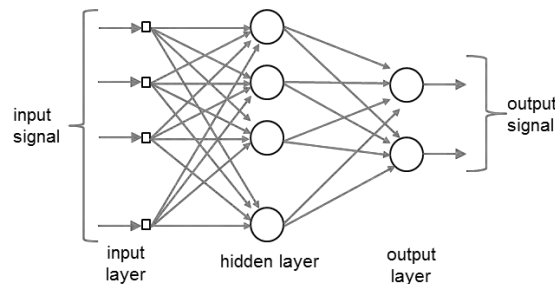

Figure 4. Multilayer perceptron

For a supervised ANN, two distinct phases can be devised. At the training phase, the connection weights are adjusted for the best performance of the network in establishing the mapping of many input/output vector pairs. The backpropagation training is a supervised learning, where pairs of input and output data are used to determine the error of the network as the square difference between the calculated output and the desired vector, for the same input. The connection weights are adjusted by an amount proportional to the error, following the so-called delta rule,

$$w_{ji}(n) = w_{ji}(n) + \alpha\left(w_{ji}(n) - w_{ji}(n-1)\right) + \Delta w_{ji}, \tag{24}$$

with

$$\Delta w_{ji}(n) = \eta\, \delta_j(n)\, y_i(n), \tag{25}$$

where α is a momentum parameter, η is the learning rate and the local gradient $\delta_j(n)$ is defined by

$$\delta_j(n) = e_j(n)\varphi_j'\left(v_j(n)\right) \tag{26}$$

where $e_j(n)$ is the calculated error. Once trained, the weights are fixed, and the run phase can be performed, where the network is presented to new inputs and calculates the corresponding outputs, based on what it has learned.

For finding the best configuration of an ANN is an exhaustive task, because there are no rules to define the optimum number of hidden layers, the number of neurons on each hidden layer, the best activation function, or the best values for learning rate (η) and momentum (α). The common procedure is to test several configurations of ANNs, evaluate each one manually and define which gave the best result.

In the last two decades, a number of authors are developing algorithms for automatically identify an optimal architecture for the ANN (Costa *et al.*, 2003; Carvalho *et al.*, 2010; Rocha, 2012). The definition of the architecture for an ANN can be formulated as an optimization problem, where each point in the search space represents one architecture. Considering some objective function, that can take into account the training error, the size of the ANN or any other parameter, the costs of all possible architectures compose a discrete surface of search, where the optimal one is that with the performance as close as possible of a global optimum.

The objective function at the present work combines two error criteria and one penalty factor, being defined as (Carvalho *et al.*, 2010)

$$J(K) = penalty \ x \ \left(\frac{p_1 \ x \ E_{train} + \ p_2 \ x \ E_{gen}}{p_1 + p_2} \right), \tag{27}$$

where p_1 and p_2 are weights that modify the relevance allowed to the training and generalization error, E_{train} and E_{gen}, both defined as

$$E = \frac{1}{N} \sum_{k=1}^{N} (d_k - y_k)^2, \tag{28}$$

where N is the number of patterns, d_k the desired output and y_k the calculated output. K is the vector of unknown parameter: number of hidden layers, number of neurons per hidden layer, type of activation function, learning and momentum ratios. The *penalty* function is used to control the ANN complexity, avoiding solutions with too much layers, or with too much neurons in each layer, but at the present work we consider $penalty = 1$. Weights p_1 and p_2 assume the values of 1 and 0.1.

The Multi-Particle Collision Algorithm (MPCA) is the metaheuristic used for solving the optimization problem given by Eq. (26). MPCA is a new metaheuristic introduced by Luz *et al.* (2008) as an extension of the Particle Collision Algorithm (PCA), proposed by Sacco and Oliveira (2005).

The PCA has been inspired on the behavior of absorption and scattering of neutrons in nuclear reactors and resembles, in structure, the simulated annealing algorithm. Firstly, in the PCA, an initial configuration is chosen; then there is a stochastic perturbation of the old configuration to generate a new one. The qualities of the two configurations are compared. If the new configuration is better, it is named as the old configuration for the next step, where will be performed an exploration of the vicinity of this solution. If it is not, the algorithm proceeds with a new change of the old configuration, called "scattering", and the whole procedures is repeated until a stop criteria is reached.

MPCA is based on canonical PCA, but introduces the use of several particles, instead of only one particle, acting in a collaborative way over the search space. It is intrinsically a parallel code, that makes the division of task by multiple particles for parallel machines of multi-cores processors. The pseudo-code for MPCA is presented by Fig. 5 (Luz *et al.*, 2008)

```
Generate an initial solution Old_Config
Best_Fitness = Fitness (Old_Config)
Update blackboard
For n = 0 to # iterations
            For n = 0 to # of particles
                        Perturbation ( )
                        If Fitness (New_Config) > Fitness (Old_Config)
                                    If Fitness (New_Config) > Best_fitness
                                                Best_fitness = Fitness (New_Config)
                                    End if
                                    Old_Config = New_Config
                                    Exploration ()
                        else
                                    Scattering ()
                        End if
                        Update blackboard
            End for
End for
```

Figure 5. Pseudo-code of MPCA

At the present work, the search space of MPCA is given by the parameters shown on Tab. 1. The MPCA is used to generate a set of candidate solutions that correspond to an ANN architecture. For each solution, the ANN is activated, and the training process runs until the stopping criteria of minimum error is satisfied. With the values obtained by ANN, the MPCA calculates the objective function, updating the parameters for the ANN. The process is repeated until an optimal value for the objective function is found.

Table 1. Parameters of the search space of MPCA

Parameter	Value
Number of hidden layers	1 to 3
Number of neurons in 1st hidden layer	1 to 32
Number of neurons in 2nd hidden layer	1 to 32
Number of neurons in 3rd hidden layer	1 to 32
Learning ratio	0.0 to 1.0
Momentum constant	0.1 to 0.9
Activation function	Logistic, Tanh, Gaussian

4. RESULTS

4.1 Methodology

To solve the proposed problem, the methodology shown schematically on Fig. 6 will be used.

A number of 646 temperature profiles were randomly selected from TIGR (Thermodynamic Initial Guess Retrieval) database. Seven wavelengths from HIRS/2 sensor of NOAA-14 satellite, associated to CO_2 band absorption, centered on 15 μm, have been used, in order to provide a reasonable vertical coverage of the most important interval to the numerical weather prediction (from sea level up to 250 hPa) (Liou, 2002). For the numerical experiments, properties for cumulus cloud, with 1 km height, were considered (I3RC, 2012; WDC-RSAT, 2012) and are represented on Tab. 2 and Tab. 3.

Table 2. Cloud coefficients for the chosen wavelengths

Wavelength (μm)	Single scattering albedo	Extinction coefficient (m^{-1})	Scattering coefficient (m^{-1})
14.95	0.346002	0.06655	0.02303
14.72	0.341669	0.06522	0.02228
14.50	0.337592	0.06398	0.02160
14.21	0.332327	0.06238	0.02073
14.00	0.32839	0.06120	0.02010
13.66	0.322336	0.05940	0.01915
13.34	0.316803	0.05777	0.01830

Table 3. Phase function coefficients

Coefficient	Value	Coefficient	Value
ω_0	1.0	ω_{10}	0.04850
ω_1	2.44421	ω_{11}	0.02046
ω_2	2.97751	ω_{12}	0.00825
ω_3	2.72090	ω_{13}	0.00319
ω_4	2.08455	ω_{14}	0.00119
ω_5	1.39373	ω_{15}	0.00043
ω_6	0.83649	ω_{16}	0.00015
ω_7	0.45720	ω_{17}	0.00005
ω_8	0.23117	ω_{18}	0.00002
ω_9	0.10904	ω_{19}	0.00001

ANN for step-2 consists of eight inputs (radiances for seven wavelengths plus the level of the atmosphere) and one output: the temperature for that level (Fig. 7). The technique of self-configured ANN has determined the best architecture for one hidden layer, with ten neurons, logistic function as activation function, learning rate of 0.430439 and momentum equals to 0.852758.
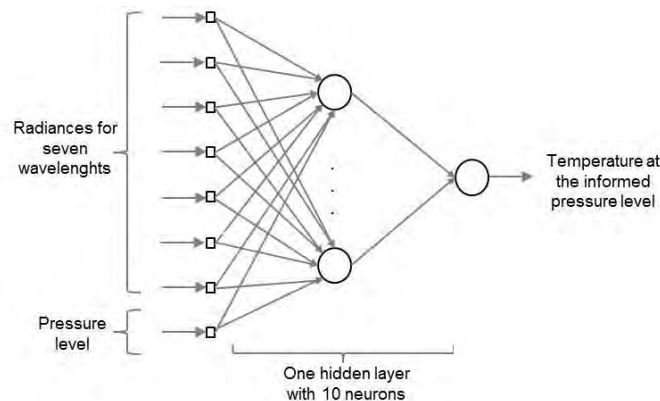

Figure 7. Scheme of the neural network used to estimate temperature profile

ANN for step-1 consists of ten inputs (radiances that leave the top of the cloud measured by satellites for ten polar directions and one wavelength) and one output: the radiance that reaches the bottom of the cloud for a given wavelength

(Fig. 8). The best architecture was an ANN with three hidden layers, with 10, 20 and 14 neurons respectively, Gaussian function as activation function, learning rate of 0.832256 and momentum equals to 0.152039.
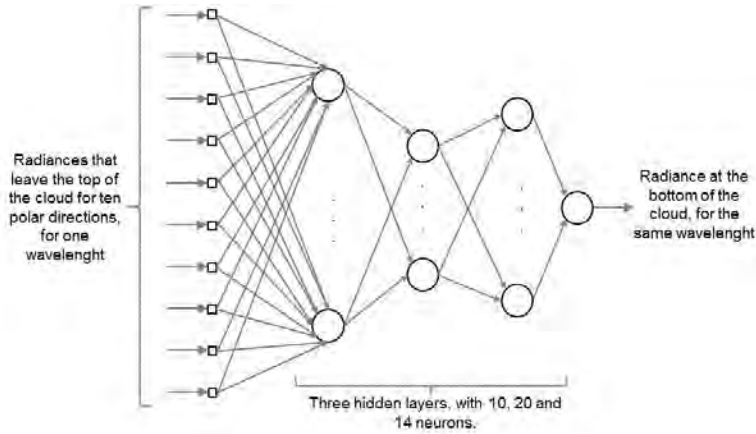

Figure 8. Scheme of the neural network used to estimate boundary condition

Once the best architecture for each ANN is determined, the whole inverse problem is solved by employing radiances data for seven wavelengths to the ANN of step-1, obtaining the radiative boundary condition for the bottom of the cloud. After that, these seven radiance values, plus the levels of interest in the atmosphere, are presented to the second ANN (step-2), to find the temperature for that level.

4.2 Preliminary results

Four test cases have been chosen, from the whole set of patterns, to test the methodology. As mentioned before, the radiances for the seven wavelengths at the bottom of the cloud are firstly estimated. The temperature profile, from 1000 hPa up to 0,1 hPa, is estimated by the second ANN. The region until 250 hPa is the most important vertical interval for numerical weather prediction, and only this interval is shown here. Figures 9 to 12 show the obtained results, and indicate the largest relative error for estimated temperature, in the region between 1000 hPa and 250 hPa. It is clear the correlation between the precision on the estimates of the radiances at the bottom of the cloud and the temperature profile estimation.
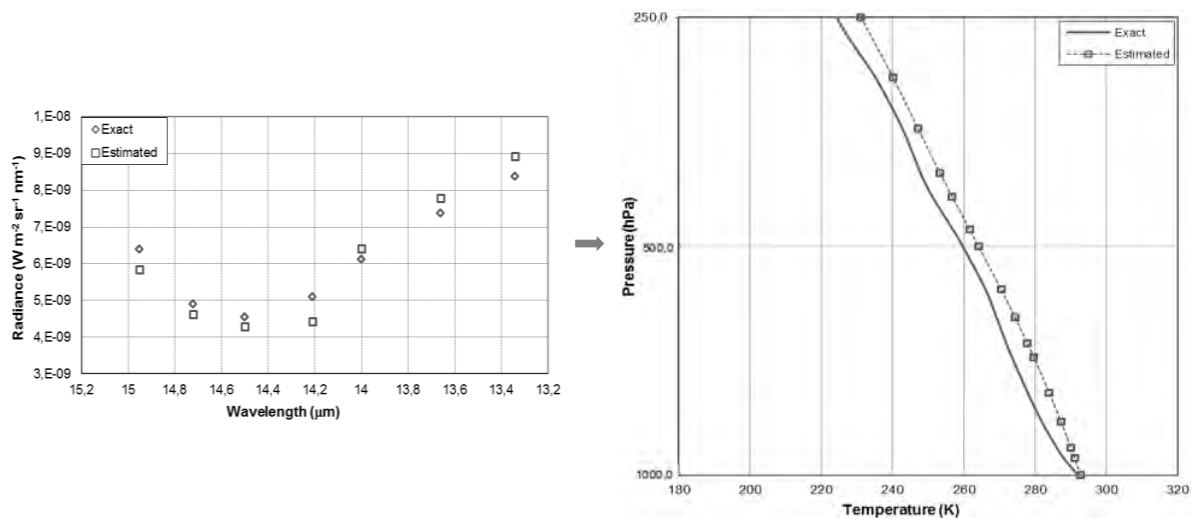

Figure 9. Results for test case 1 – Largest relative error: 3%
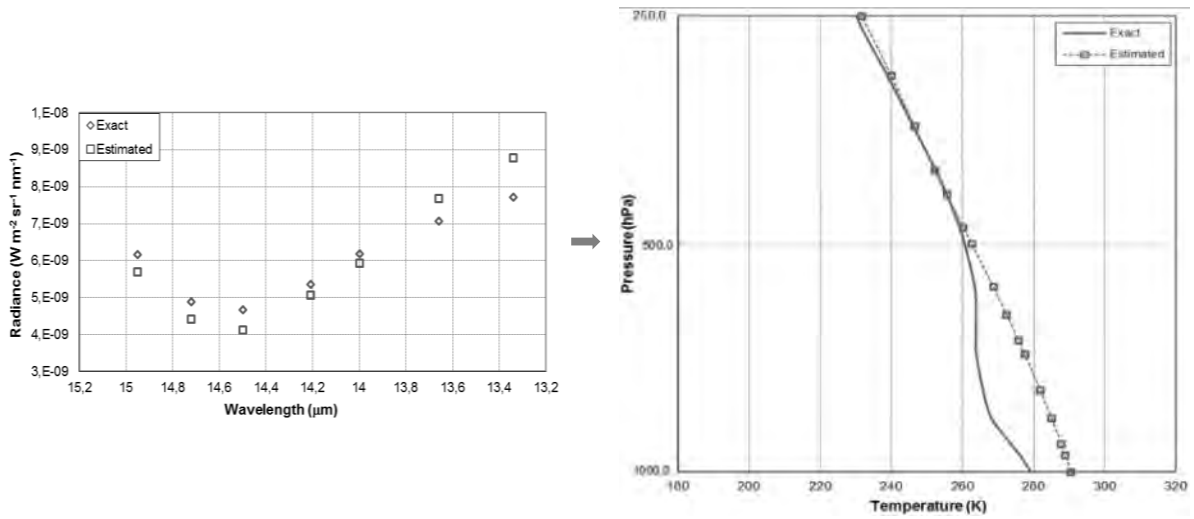
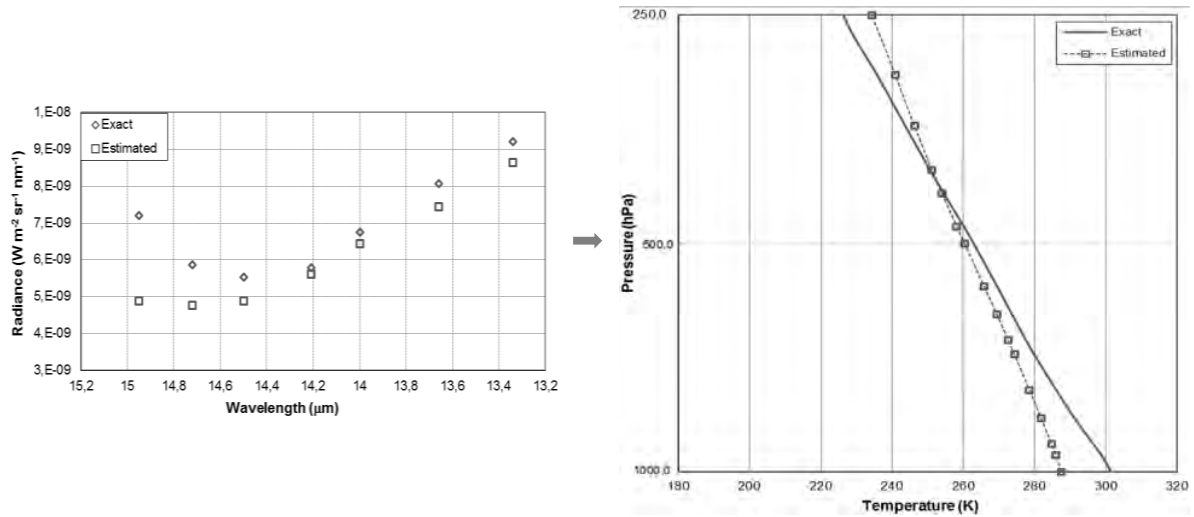Figure 10. Results for test case 2 – Largest relative error: 6.3%

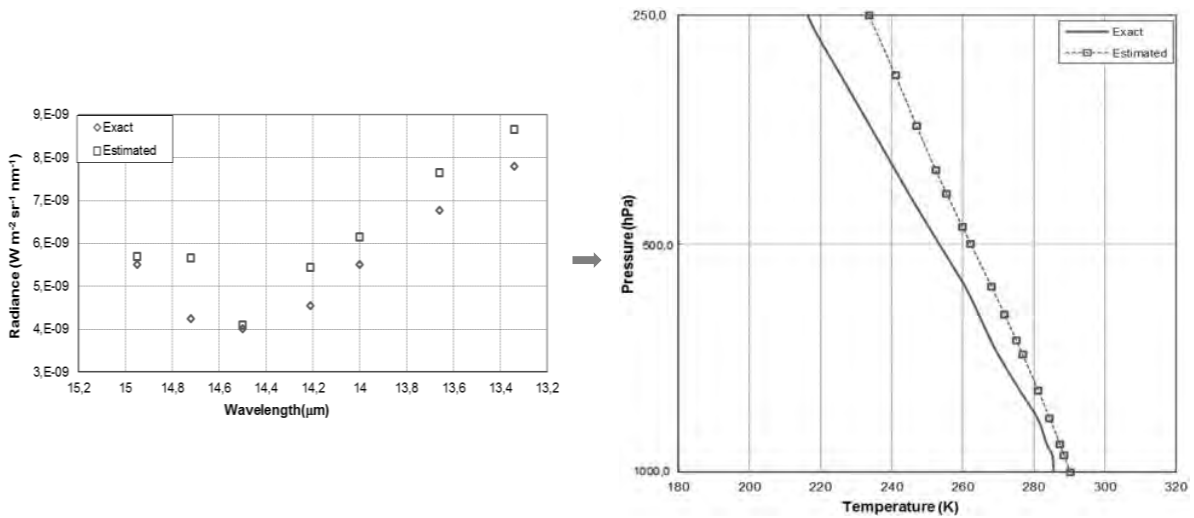Figure 11. Results for test case 3 – Largest relative error: 4.6%

Figure 12. Results for test case 4 – Largest relative error: 8%

5. CONCLUSIONS

A methodology based on self-configured artificial neural networks to retrieve atmospheric temperature profile from satellite data is performed under cloud covering. These preliminary results show that the proposed technique produces satisfactory results, and it has great potential of use. Such algorithm allows for obtaining information from images covered by thick clouds, where not either sign of microwave is able to supply reliable inversions. The results encourage us to apply the strategy out lined here in more realistic experiments.

6. REFERENCES

Barichello, L.B. and Vilhena, M.T., 1993. "A general approach to one-group one-dimensional transport equation", *Kerntechnik*, vol. 58 (3), p. 182.

Campos Velho, H.F., Retamoso, M.R., Vilhena, M.T., 2002. "Inverse problems for estimating bottom boundary conditions of natural waters". *Int. J. Num. Methods Eng.*, vol. 54 (9), p. 1357.

Carvalho, A., Ramos, F.M., Chaves, A.A., 2010. "Metaheuristics for the feedforward artificial neural network (ANN) architecture optimization problem". *Neural Comp. Applic.*, vol. 20 (8), p. 1273.

Costa, M.A., Braga, A.P., Menezes, B.R., 2003. "Improving neural networks generalization with new constructive and pruning methods". *Intelligent and Fuzzy Systems*, vol. 13, p. 75.

Haltiner, G.J. and Willias, R.T., 1980. *Numerical Prediction and Dynamic Meteorology*, John Wiley & Sons, 2nd edition.

Haykin, S., 2001. *Redes Neurais: princípios e prática*, Bookman, Porto Alegre, Brazil.

I3RC – Intercomparison of 3D Radiation Code, 2012. <http://i3rc.gsfc.nasa.gov/input/heatingrate.html>.

Lenoble, J., 1993. *Atmospheric radiative transfer*, A. DEEPAK Publishing, Virginia.

Liou, K-N., 2002. *An introduction to atmospheric radiation*, Academic Press, Oxford, UK.

Luz, E.F.P., Becceneri, J.C., Campos Velho, H.F., 2008. "A new multi-particle collision algorithm for optimization in a high performance environment". *J. Comp. Interdisciplinary Sci.*, vol. 1, p. 1.

Lynch, P., 2006. *The Emergence of Numerical Weather Prediction: Richardson's Dream*, Cambridge University Press.

Özişik, M.N., 1973. *Radiative heat transfer and interactions with conduction and convection*, John Wiley & Sons.

Planck, M., 1901. "Ueber das Gesetz der Energieverteilung im Normalspectrum", *Annalen der Physik*, vol. 4, p. 553.

Rocha, H.P., 2012. *Abordagens multi-objetivo para o treinamento de redes neurais e seleção de características*. Dissertation (Master on Electric Engineer), Federal University of Minas Gerais, Belo Horizonte, Brazil.

Richardson, L.F., 1922. *Weather Prediction by Numerical Process*, Cambridge University Press.

Sacco, W.F. and Oliveira, C.R.E.A., 2005. "A new stochastic optimization algorithm based on particle collision metaheuristic". In: *6th WCSMO - World Congress of Structural and Multidisciplinary Optimization, WCSMO*, Rio de Janeiro, Brazil.

Shiguemori, E.H., Silva, J.D.S., Campos Velho, H.F., Carvalho, J.C., 2006. "Neural network based models in the inversion of temperature vertical profiles from radiation data". *Inverse Prob. Sci. Eng.*, vol. 14 (5), p. 543.

Souto, R.P., Campos Velho, H.F., Stephany, S., 2006. "Reconstruction of vertical profiles of the absorption and scattering coefficients from multispectral radiances". *Mathematics and Computers in Simulation*, vol. 73, p. 255.

WDC-RSAT, 2012. OPAC – Optical Properties of Aerosols and Clouds. <http://wdc.dlr.de/data_products/SPECTRA/opac/forms/>.

7. RESPONSIBILITY NOTICE

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