# Investigation of methodologies for atmospheric retrieval for the CPTEC operational system

**Elcio H. Shiguemori[1], Rodrigo A. F. de Souza[2], Haroldo F. de Campos Velho[1], José Demisio S. da Silva[1], Wagner Flauber A. Lima[2], João C. Carvalho[3]**

[1] *Laboratory for Computing and Applied Mathematics (LAC) – National Institute for Space Research (INPE) – São José dos Campos – SP – Brazil*
[2] *Division of Satellites and Environmental Systems / Center for Weather Forecasting and Climate Studies – DSA/CPTEC – INPE – Cachoeira Paulista – SP – Brazil*
[3] *Earth Observation (OBT) – INPE – São José dos Campos – SP - Brazil*

## Abstract

The Center for Weather Forecasting and Climate Studies (CPTEC) is responsible for producing weather maps for the numerical prediction in Brazil. One key issue for numerical prediction is related to provide good estimation of the initial conditions for the atmospheric simulation code. One procedure consists of retrieving vertical atmospheric profiles for temperature and moisture profiles. The CPTEC operationally uses the Inversion Coupled with Imager (ICI-3) software in dynamic mode (CPTEC analysis) with the ATOVS/NOAA-16 system to supply such vertical profiles. However, CPTEC is also investigating new retrieval schemes that that have been developed at INPE. One of these schemes retrieves the profiles by means of a generalized least square problem, where a new regularization operator is employed. Such regularization operator is based on maximum entropy of second order. An Artificial Neural Network (ANN) is another scheme for retrieving the atmospheric profiles. The ANN is the Multi-layer Perceptron, with back propagation learning strategy. The goal here is to compare these different methods, focusing on the operational procedures. The comparison is carried out by using two databases: TIGR and NESDISPR. About 500 profiles from the TIGR and 400 profiles from the NESDISPR, and associated radiances, are selected for testing the three strategies. The average error over profiles is used to perform the comparison among the inversion methodologies, and these analyses will be shown here.

## Introduction

Vertical profiles of the temperature and water vapor measurements are fundamental for the meteorological process of the atmosphere. The monitoring of these quantities required observational stations all over the world, however logistics and economic problems lead to a lack of sensors in many parts of the planet. Thus, retrieving temperature and humidity profiles from satellite radiance data became very important for weather analyses and data assimilation processes in numerical weather predictions models.

Satellite measured radiance data may be interpreted by the inversion of the Radiative Transfer Equation (RTE) that relates the measured radiation in different frequencies to the energy from different atmospheric regions. The degree of indetermination is associated with the spectral resolution and the number of spectral channels, and this solution is usually very unstable due to noise in the measuring processes (Rodgers 1976, Twomey 1977). Several methodologies and models have been developed attempting at improving data processing for information extraction from satellite radiance data (Chahine 1970, Liou 1982, Smith et al. 1985).

The atmospheric temperature estimation is a classical inverse problem. In order to deal with the ill-posed characteristic of the inverse problem, regularized solutions (Tikhonov and Arsenin 1977,

Ramos and Campos Velho, 1996; Campos Velho and Ramos, 1997; Ramos et al., 1999) and also regularized iterative solutions (Alifanov, 1974; Jarny et al.; 1991; Chiwiacowsky and Campos Velho, 2003) have been proposed. Recently, artificial neural networks have also been employed for solving inverse problems (Aires et al., 2002; Atalla and Inman, 1998; Hidalgo and Gómez-Treviño, 1996; Krejsa et al., 1999; Woodbury, 2000; Shiguemori et al., 2004). In this paper, a MLP (Multi Layer Perceptron) Artificial Neural Network (ANN) is used to address this problem.

The supervised learning nature of the MLP network requires training sets to be furnished as inputs and desired outputs. The inputs are measured satellite radiances in different spectral channels, and the outputs are the desired corresponding absolute temperature profiles obtained by solving the forward model. The MLP network was trained with TIGR temperature profiles database (Chédin et al., 1995; TIGR 2005), and NESDISPR database obtained by solving the forward model. Generalization tests used TIGR and NESDISPR database examples, in which they were not used in training phase.

Experiments using both methodologies are employed considering the High Resolution Radiation Sounder (HIRS) of NOAA-16 satellite. HIRS is one of the three sounding instruments of the TIROS Operational Vertical Sounder (TOVS).

In this work, the forward model was used in the entropic in the objective function (Carvalho et al 1999, Ramos et al. 1999). In the ANN, the RTTOVS-7 model was used to generate the train data set, and also for validation and generalization databases. This model simulates the radiances of HIRS-2 NOAA-16 satellite.

## Forward Model

Equation below represents the mathematical formulation of the forward problem that permits the calculation of radiance values from associated temperatures (Liou, 1982)

$$I_\lambda(0) = B_\lambda(T_s)\Im_\lambda(p_s) + \int_{P_s}^{0} B_{\lambda i}[T(p)]\frac{\partial \Im_\lambda(p)}{\partial p}dp \qquad (1)$$

where, $I_\lambda$ is the value of the spectral radiance, subscript $s$ denotes surface; $\lambda$ is the channel wavelength; $p$ is the considered pressure; $\Im$ is the space atmospheric layer transmittance function that is a function of the wavelength and the concentration of absorbent gas, which usually declines exponentially with the height. In pressure coordinate, the transmittance may be expressed by:

$$\Im_\lambda(p) = \exp\left[-\frac{1}{g}\int_{p_0}^{p} k_\nu(p)q(p)dp\right] \qquad (2)$$

where, $k_\nu$ is the absorption coefficient; $q$ is the ratio of gas mixture; $g$ is the acceleration of the local gravity; and $p_0$ is the pressure in the top of atmosphere; $B$ is Planck's function (equation 3), which is a function of the temperature $T$ and the wavelength $\lambda$:

$$B_\lambda(T) = \frac{2hc^2/\lambda^5}{[e^{hc/k_B\lambda T}-1]} \qquad (3)$$

$h$ is Planck's constant; $c$ is the light speed; and $k_B$ is Boltzmann's constant.

Equation (1) is discretized using central finite differences leading to equation:

$$I_i(0) = B_{i,s}(T_s)\mathfrak{I}_{i,s} + \sum_{j=1}^{N_p}\left(\frac{B_{i,j} + B_{i,j-1}}{2}\right)\left[\mathfrak{I}_{i,j} - \mathfrak{I}_{i,j-1}\right]$$ (4)

where $i = 1, \ldots, N_\lambda$, $I_i \equiv I_{\lambda_i}$; $N_\lambda$ is the number of satellite channels; $B_{i,j} = B_{\lambda_i}(T_j)$; $\mathfrak{I}_{i,j} = \mathfrak{I}_{\lambda_i}(p_j)$; and $N_p$ is the number of atmospheric layers considered. It is assumed that each atmospheric layer has a characteristic temperature $T_j$ to be computed.

In this work, the forward model is used to the both methodologies. The entropic regularization employs it in the minimization functional and the ANN to simulate radiances to train, validation and generalization phases, its necessary because there are no satellite radiances and radiosonde data.

## Entropic Regularization

The inverse problem is assumed to be solved defined as follows (Ramos and Campos Velho, 1996; Campos Velho and Ramos 1997, Carvalho et al. 1999):

find $T$ such that $I = K(T)$ (5)

where $T \in \mathfrak{R}^n$ denotes the unknown parameters, $I \in \mathfrak{R}^n$ the data-vector and $K : \mathfrak{R}^n \rightarrow \mathfrak{R}^m$ is an operator, linear or not, modeling the relation between $T$ and $I$. Here, the mathetical model $K(.)$ is expressed by Equation (4).

A traditional approach for solving Equation (5) is to determine $T$ in the least square sense. Unfortunately, minimization of the distance between computed and experimental data alone does not provide a safe inversion technique, due to the presence of noise in y. A better approach, is to formulate the inverse problem as:

$$\min\left\{\left\|I^{Sat} - I^{Mod}(T)\right\|_2^2 + \alpha\Omega[T]\right\}$$ (6)

where $\|\cdot\|$ is a suitable norm, usually the Euclidean square norm, $\Omega$ is a regularization operator, and $\alpha$ is the regularization parameter. The operator $\Omega(t)$ generally expresses *a priori* information about the unknown physical model. In the case of Maximum Entropy regularization, $\Omega(T)$ takes the form of Shannon's missing information measure:

$$\Omega(T) = S(T) = -\sum_{i=1}^{N} q_i \log q_i, \qquad q_i = \frac{p_i}{\sum_{i=1}^{n} p_i}$$ (7)

$S(T)$ attain its global maximum when all $q_i$ are the same, which corresponds to a uniform distribution with a value of $S_{max} = \log N$. On the other hand, the lowest entropy level, $S_{min} = 0$, is attained when all elements $q_i$ but one are set to zero. Maximum Entropy regularization selects the simplest possible solution, containing the minimum of structure required to fit the data.

Ramos and Campos Velho (1996) – see also Campos Velho and Ramos (1997), and Carvalho et al. (1999) – proposed a generalization of the standard Maximum Entropy regularization method, which allows for a greater exibility when introducing prior information about the expected structure of

the true physical model - or its derivatives - into the inversion procedure. The entropic regularization function is defined as follows:

$$S(x) = -\sum_{i=1}^{n} q_i \log q_i, \qquad q_i = \frac{p_i}{\sum_{i=1}^{n} p_i} \tag{8}$$

$$\mathbf{p} = \Delta^{\alpha} \mathbf{T} \tag{9}$$

where $\alpha = 0, 1, 2, \ldots$ and $\Delta$ is a discrete difference operator.

A method, denoted MaxEnt-2, for a second order entropic regularization is expressed by (Ramos and Campos Velho, 1996; Campos Velho and Ramos 1997, Carvalho et al. 1999)

$$p_i = t_{i+1} - 2t_i + t_{x-1} + 2(t_{max} - t_{min}) + \zeta, \quad \text{with} \quad i = 2, \ldots, n-1. \tag{10}$$

In this work, the optimization problem is iteratively solved by the quasi-newtonian optimizer routine from the NAG Fortran Library (E04UCF 1995), with variable metrics. This algorithm is designed to minimize an arbitrary smooth function subject to constraints (simple bound, linear or nonlinear constraints), using a sequential programming method. This routine has been successfully used in several previous works: in geophysics, hydrologic optics, and meteorology.

## Artificial Neural Networks

Artificial Neural Networks (ANN) techniques have become important tools for information processing. Properties of ANNs make them appropriate for application in pattern recognition, signal processing, image processing, financing, computer vision, and so on. There are several ANN different architectures. Here a Multilayer Perceptron (MLP) with backpropagation learning is employed.

Artificial neural networks are made of arrangements of processing elements (neurons). The artificial neuron model basically consists of a linear combiner followed by an activation function, Figure 1 (a), given by:

$$y_k = \varphi\left(\sum_{j=1}^{n} w_{kj} x_j + b_k\right) \tag{11}$$

where $w_{kj}$ are the connections weights, $b_k$ is a threshold parameter, $x_j$ is the input vector and $y_k$ is the output of the $k^{\text{th}}$ neuron, the $\varphi$ is the function that provides the activation for the neuron. Neural networks will solve nonlinear problems, if nonlinear activation functions are used for the hidden and/or the output layers.

Arrangements of such units form the ANNs that are characterized by: 1. Very simple neuron-like processing elements; 2. Weighted connections between the processing elements (where knowledge is stored); 3. Highly parallel processing and distributed control; 4. Automatic learning of internal representations. ANNs aim to explore the massively parallel network of simple elements in order to yield a result in a very short time slice and, at the same time, with insensitivity to loss and failure of some of the elements of the network. These properties make artificial neural networks appropriate for application in pattern recognition, signal processing, image processing, financing, computer vision, engineering, etc. (Haykin 1994).

There exist different ANN architectures that are dependent upon the learning strategy adopted. This paper briefly describes the one ANN used in our simulations: the multilayer Perceptron with backpropagation learning. Detailed introduction on ANNs can be found in (Haykin 1994).

Multilayer perceptrons with backpropagation learning algorithm, commonly referred to as backpropagation neural networks are feedforward networks composed of an input layer, an output layer, and a number of hidden layers, whose aim is to extract high order statistics from the input data.

Figure 1 (b) depicts a backpropagation neural network with a hidden layer. A feedforward network can input vectors of real values onto output vector of real values. The connections among the several neurons have associated weights that are adjusted during the learning process, thus changing the performance of the network. Two distinct phases can be devised while using an ANN: the training phase (learning process) and the run phase (activation of the network). The training phase consists of adjusting the weights for the best performance of the network in establishing the mapping of many input/output vector pairs. Once trained, the weights are fixed and the network can be presented to new inputs for which it calculates the corresponding outputs, based on what it has learned.
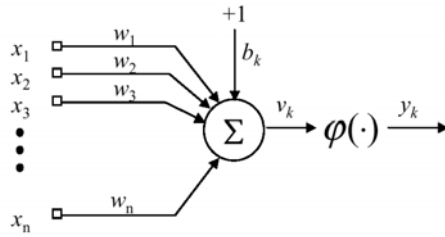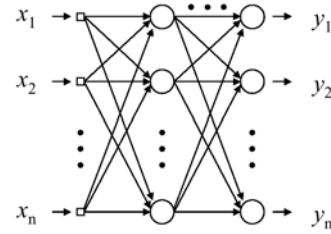


Figure 1 (a) Single Neuron            Figure 1 (b) Multilayer Neural Network

The backpropagation training is a supervised learning algorithm that requires both input and output (desired) data. Such pairs permit the calculation of the error of the network as the difference between the calculated output and the desired vector. The weight adjustments are conducted by backpropagating such error to the network, governed by a change rule. The weights are changed by an amount proportional to the error at that unit, times the output of the unit feeding into the weight. Equation (xx) shows the general weight correction according to the so-called the delta rule

$$\Delta w_{kj} = \eta \delta_k y_j \tag{12}$$

where, $\delta_k$ is the local gradient, $y_k$ is the input signal of neuron $j$, and $\eta$ is the learning rate parameter that controls the strength of change.

## Retrieval using ANN

Artificial neural networks have two stages in their application, the learning and activation steps. During the learning step, the weights and bias corresponding to each connection are adjusted to some reference examples. For activation, the output is obtained based on the weights and bias computed in the learning phase. Its important to test the ANN with data not used in training phase, this process is called generalization phase.

## Noise data simulation

The experimental data, which intrinsically contains errors in the real world, is simulated by adding a random perturbation to the exact solution of the direct problem, such that

$$\tilde{I} = I_{\text{exact}} + I_{\text{exact}}\sigma\mu \tag{13}$$

where $\sigma$ is the standard deviation of the noise and $\mu$ is a random variable taken from a Gaussian distribution. The noise data was based in actual situation of NOAA-16, HIRS Instrument, Longwave Channel Noise (NOAA-16 2005).

## Databases

Two different database for temperature profiles were used: TIGR (Thermodynamic Initial Guess Retrieval) and NESDISPR worldwide climatological profile - a file created by the NOAA/NESDIS. The data sets were divided as: training, validation, and generalization data sets (Figure 2). The training and validation data are used for training the Artificial Neural Network, and the generalization data set are profiles not used during the training phase. The Table (1) show the number of temperature profiles of each database.
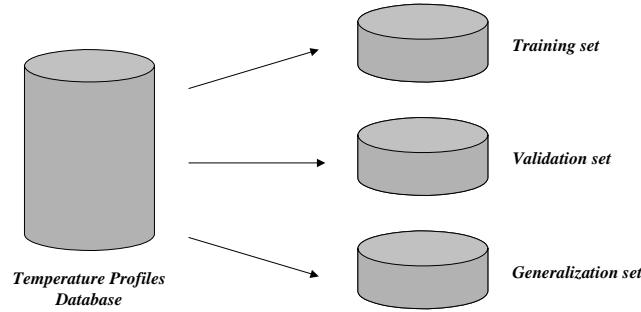


Figure 2 – Database

Table 1 – Number of profiles.

| Database | Training | Validation | Generalization |
|----------|----------|------------|----------------|
| TIGR | 587 | 587 | 587 |
| NESDISPR | 405 | 400 | 400 |

## Results analysis

The results presented in this paper were obtained using a code in Fortran 90 for implementing the artificial neural network and maximum entropy regularization. There are very good computer packages for neural networks available, but we decide to have our code, because we are looking at an operational application.

The root mean-squared error of the generalization sets TIGR (587 profiles) and NESDISPR (400 profiles) are presented in Table 3, the errors are calculated in the Layer-1: 0.1 up to 15 hPa; Layer 2: 20 up to 70 hPa; Layer-3: 85 up to 200 hPa; Layer-4: 250 up to 475 hPa; Layer-5: 500 up to 1000 hPa; This segmentation feature is important because the main interest for meteorological purposes is in the layers below 100 hPa, , where 1 hPa = 100 Pa.

The error for each Layer is computed by:

$$ASE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(T_i^{\text{Exact}} - T_i^{\text{Estimated}})^2} \tag{14}$$
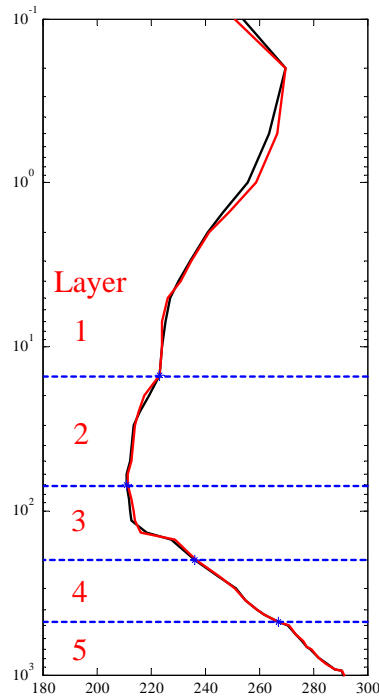
where $N$ is the number of sample points at each layer.



Figure 3 – Layers of atmospheric profile

## Entropic Regularization Results

Entropic regularized solution is obtained by choosing the function $t*$ that minimizes the functional (2). Table 2 presents the average error of entropic regularization (MaxEnt-2), defined by Eq. (14), of TIGR (587 profiles) and NESDISPR (400 profiles) databases. It was used a mean regularization parameter ($\approx 5.0$) for all cases. It can be noted that the error for the Layer 1 and Layer 2 have good approximation, these layers are more important for meteorological purposes. The Figures 3 (a-b) present two examples of TIGR database and Figures 3 (c-d) NESDISPR database examples.
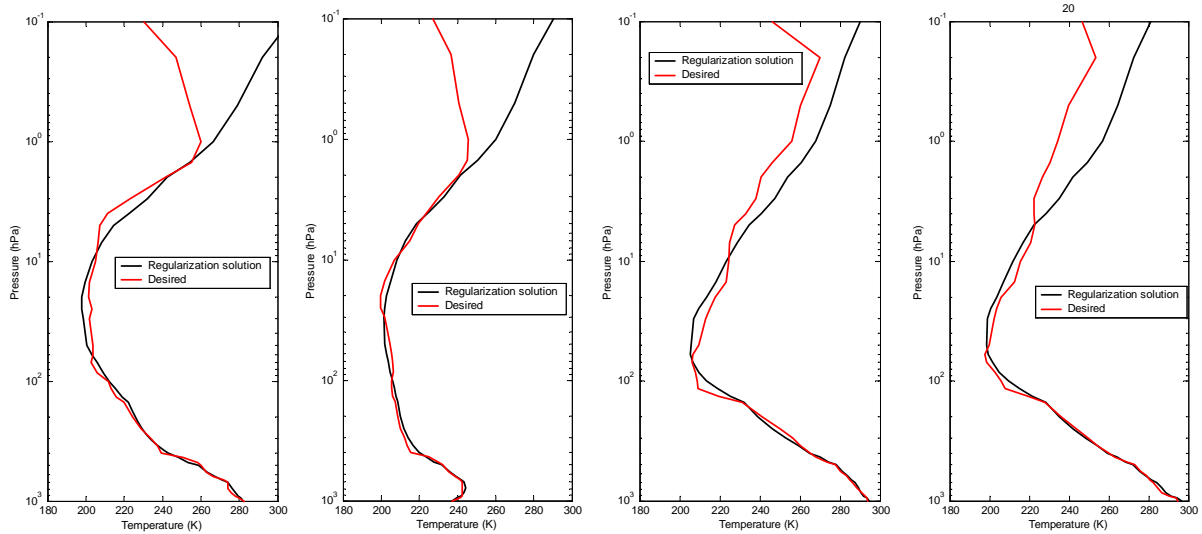


Figure 4 Examples results obtained with second order entropic regularization (a) and (b) TIGR database. (c) and (d) – NESDISPR database.

## Artificial Neural Networks Results

In the activation phase, the inverse problem is solved by weights and bias obtained during the training phase. The robustness of the trained MLP is evaluated employing satellite radiances and temperature profiles not used in the training phase (generalization).

The generalization capacity of the MLP is verified considering TIGR (587 profiles) and NESDISPR (400 profiles) databases. Table 2 shows the ANN average error (ANN) of generalization databases, defined by Eq. (14), using 8 hidden neurons in the hidden layer. The estimation using ANN presents small errors. The following figures present some examples of the ANN results. The Figure 5 (a-b) were obtained by the ANN trained with TIGR database. The Figure 4 (c-d) with the ANN trained with NESDISPR database.
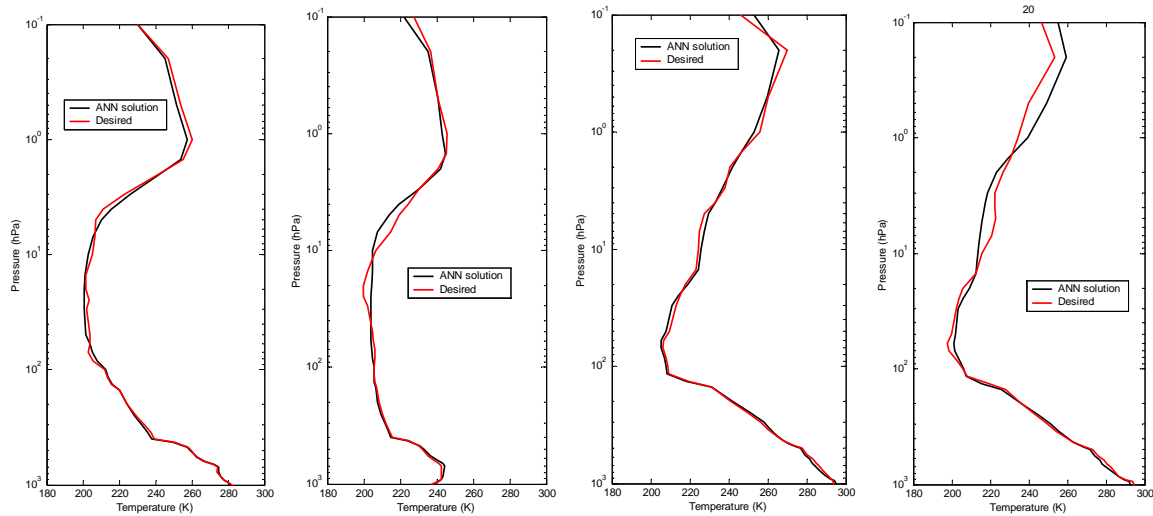


Figure 5 (a) and (b) Examples of generalization tests, obtained with the ANN trained with TIGR data sets. (c) and (d) – Generalization tests, obtained with the ANN trained with NESDISPR data sets.

Table 2 – Error for generalization database.

| Methodology | Database | Neurons | Layer 1 | Layer 2 | Layer 3 | Layer 4 | Layer 5 |
|---|---|---|---|---|---|---|---|
| *ANN* | *TIGR* | *8* | *1.8161* | *0.9438* | *0.7043* | *0.7308* | *0.5153* |
| MaxEnt-2 | TIGR | * | 7.2910 | 1.4441 | 1.1592 | 1.5244 | 0.6869 |
| *ANN* | *NESDISPR* | *8* | *0.8836* | *0.6641* | *0.7714* | *0.3720* | *0.4849* |
| MaxEnt-2 | NESDIS | * | 5.0706 | 1.5024 | 2.2225 | 0.9468 | 0.5057 |

## Final Remarks

The retrieval of atmospheric temperature profile has been addressed using two different approaches, and employing two temperature databases: TIGR and NESDISPR, testing about 1000 profiles. Both methodologies implemented have produced good inversions, even for data containing real noise of the NOAA-16 satellite HIRS sounder.

Concerning the computational time requested by the methodologies used in this work, it should be pointed out that the ANN have two different phases: training and activation. The training phase usually is very CPU time consuming, and for the present problem it is requiring some hours. However, this step is done only one time. After the training, the activation phase is very fast, usually takes less than one second. The latter phase represents the real inverse problem solution. Regarding

the regularization, the CPU-time was around few seconds. All computational simulations have been performed in a personal computer with an Athlon - 1.8 GHz processor.

## Acknowledgments

## References

Aires, F., Chedin, A., Scott, N.A. and Rossow, W.B. 2002. A regularized neural net approach for retrieval of atmospheric and surface temperatures with the IASI instrument. *J. of App. Meteorology*, **41** (2): 144-159.

Alifanov, O.M. 1974. Solution of an inverse problem of heat conduction by iteration methods. *Journal of Engineering Physics*, **26** (4): 471-476.

Atalla, M.J. and D.J. Inman, On model updating using neural networks. Mechanical Systems and Signal Processing 1998; **12** (1): 135-161.

Campos Velho, H.F. and Ramos, F.M. 1997. Numerical inversion of two-dimensional geoelectric conductivity distributions from eletromagnetic ground data, *Braz. J. Geophys*., **15**: 133.

Carvalho, J.C., Ramos, F.M., Ferreira, N.J. and Campos Velho H.F. 1999. Retrieval of Vertical Temperature Profiles in the Atmosphere, *Inverse Problems in Engineering* (3ICIPE), Proc. In CD-ROM, paper code HT02 - Proc. Book: pp. 235-238, Port Ludlow, Washington, USA, June 13-18, UEF-ASME.

Chahine, M. T. 1970. Inverse Problem in Radiative Transfer: determination of atmospheric parameters, *Jour. Atmos. Sci*., **27**, 960.

Chédin A., Scott, N. A., Wahiche, C. and Moulini P. 1985. The improved initialization inversion method: A high resolution physical method for temperature retrievals from TIROS-N series. J. Climate Appl. Meteor, **24**, 128-143.

Chiwiacowsky, L. D. and Campos Velho, H.F. 2003. Different Approaches for the Solution of a Backward Heat Conduction Problem. *Inverse Problems in Engineering*, **11** (6): 471-494.

E04UCF routine. 1995. NAG Fortran Library Mark 17, Oxford, UK.

Haykin, S. 1994. Neural Networks: A Comprehensive Foundation, Macmillan. New York.

Hidalgo, H. and Gómez-Treviño, E. 1996. Application of Constructive Learning Algorithms to the Inverse Problem, *IEEE T. Geosci. Remote,* **34,** 874

Jarny, Y., Ozisik, M.N. and Bardon, J.P. 1991. A general optimization method using adjoint equation for solving multidimensional inverse heat conduction. *International Journal of Heat and Mass Transfer*, **34** (11): 2911-2919.

Krejsa, J., Woodbury, K.A., Ratliff, J.D. and Raudensky, M. 1999. Assessment of strategies and potential for neural networks in the IHCP, *Inverse Probl. Eng.*, **7**, 197.

NOAA-16 2005. available in the internet: http://www.osdpd.noaa.gov/PSB/PPP/notices/ notice_files/dec3004.html

Liou, K. N. 1982. An introduction to atmospheric radiation, academic press, Orlando.

Ramos, F.M. and Campos Velho, H.F. 1996. Reconstruction of geoelectric conductivity distributions using a minimum first-order entropy technique, *2nd International Conference on Inverse Problems on Engineering*, Le Croisic, France; **2,** 199-206.

Ramos F.M., Campos Velho H.F., Carvalho, J.C., Ferreira, N.J. 1999. Novel Approaches on Entropic Regularization, *Inverse Problems*, **15**(5), 1139-1148.

Rodgers, C. D. 1976. Retrieval of the atmospheric temperature and composition from remote measurements of thermal radiation, *Rev. Geophys. Space Phys.*, **14**, 609-624.

Shiguemori, E.H., Silva, J.D.S., Campos Velho, H.F. and Carvalho, J.C. 2004. Neural Network based Models in the Inversion of Temperature Vertical Profiles from Satellite Data, *Inverse Problems, Design and Optimization Symposium (IPDO)*, 17-19 March, Rio de Janeiro (RJ), Brazil - Proc. in CD-Rom: paper code IPDO-077.

Smith, W. L., Woolf, H. M. and Schriener, A. J. 1985. Simultaneous retrieval os surface and atmospheric parameters: a physical analytically direct approach, Adv. In Rem. Sens., **7**.

TIGR: Thermodynamic Initial Guess Retrieval. 2005. data set available in the internet: http://ara.lmd.polytechnique.fr/htdocs-public/products/TIGR/TIGR.htm - access at June 10,

Tikhonov A.N. and Arsenin V.I. 1977. Solutions of Ill-posed Problems, John Wiley & Sons.

Twomey, S. 1977. Introduction to the mathematics of inversion in remote sensing and interative measurements, Amsterdam, Elsevier Scientific.

Woodbury, K.A. 2000. Neural networks and genetic algorithms in the solution of inverse problems, *Bulletim of the Braz. Soc. for Comp. Appl. Math. (SBMAC)*, available in the internet: www.sbmac.org.br/publicacoes.