Reprinted from the preprint volume of the ***First Conference on Artificial Intelligence****, 11-16 January 1998, Phoenix, AZ, by the American Meteorological Society, Boston, MA.*

2.4 NEURAL NETWORKS AS A GENERIC TOOL FOR SATELLITE RETRIEVAL ALGORITHM DEVELOPMENT AND FOR DIRECT ASSIMILATION OF SATELLITE DATA INTO NUMERICAL MODELS.

V.M. Krasnopolsky,
Environmental Modeling Center, NCEP, NOAA[1]

1. INTRODUCTION: DERIVING GEOPHYSICAL PARAMETERS FROM SATELLITE MEASUREMENTS

Satellite remote sensing data are used by numerical weather prediction (NWP), field meteorology, fisheries, Coast Guard, the oil industry, the Navy and others. Users work with geophysical parameters such as pressure, temperature, wind speed and direction, water vapor, etc. Satellite sensors generate measurements in terms of radiances, sigma naughts, brightness temperatures, etc. Satellite retrieval algorithms which transform satellite measurements into geophysical parameters play the role of mediator between satellite measurements and users.

Conventional methods for using satellite data (standard retrievals) involve solving an inverse (or retrieval) problem and deriving a transfer function (TF), ***f***, which relates a geophysical parameter of interest, *g* (e.g., surface wind speed over the ocean), to a satellite measurement, *s* (e.g., SSM/I brightness temperatures),

$$g = \mathbf{f}(s) \tag{1}$$

where both *g* and *s* may be vectors. The TF, ***f***, may be derived explicitly or assumed implicitly. Standard retrievals have the same spatial resolution as the sensor measurements and produce instantaneous values of geophysical parameters over the areas where the measurements are available. Geophysical parameters derived using standard retrievals can be used for many applications, for example, in NWP data assimilation systems. In this case, a contribution to the analysis cost function χ_g from a particular retrieval, g^0, is:

$$\chi_g = \frac{1}{2}(g - g^O)^T (O + E)^{-1}(g - g^O) \tag{2}$$

where $g^o = \mathbf{f}(s^o)$ is retrieved geophysical parameter (s^o - a sensor measurement), *g* - value of this geophysical parameter in analysis; *O* and *E* - expected error covariance of the observations and of the retrieval algorithm. Because standard retrievals are based on solution of inverse problem which is usually mathematically ill-posed, it has some rather subtle properties and error characteristics (Eyre, 1987), which cause additional errors and problems in retrievals (e.g., amplification of errors, ambiguities, etc.). As a result, high-quality sensor measurements are converted into lower-quality geophysical parameters.

This type of errors can be avoided or reduced, using variational retrievals (or inversion) through direct assimilation of satellite measurements (Lorenc, 1986; Parrish and Derber, 1992; Phalippou, 1996; Prigent et al., 1997).

In this case, due to direct assimilation of sensor measurements, the entire data assimilation system is used for inversion (as a retrieval algorithm). In this case, a contribution to the analysis cost function χ_s from a particular sensor measurement, s^0, is:

$$\chi_s = \frac{1}{2}(s - s^O)^T (O + E)^{-1}(s - s^O) \tag{3}$$

where

$$s = F(g) \tag{4}$$

F is a forward model (FM) which relates an analysis state vector *g* (vector of geophysical parameters in analysis) to a vector of simulated sensor measurements, *s*; *O* and *E* - expected error covariance of the observations and of the forward model. The retrieval in this case is an entire field(global in the case of the global data assimilation system) for the geophysical parameter *g* which has the same resolution as the numerical model used in the data assimilation system. This resolution may be lower than the resolution of standard retrievals. The variational retrievals are also not instantaneous but averaged in time over the analysis cycle (several hours); however, the field is continuous and coherent. It is important to emphasis one very significant difference between using the TF for standard retrievals and the FM in variational retrievals. In standard retrievals the TF (1) is applied one time per sensor observation to produce a geophysical retrieval. In variational retrievals the FM and its partial derivatives (the number of derivatives is equal to $m \times n$, where *m* and *n* are the dimensions of the vectors *g* and *s* respectively) have to be estimated for each of *k* iterations performed during minimization of the cost function (3), that is $(m \times n + 1) \times k$ times (e.g., about 3000 times for SSM/I). Taking into account that an FM is often much more complicated than a TF, the requirements for simplicity of the FM used in the variational retrievals are very restrictive,

[1]*Coresponding author address:* Dr. V. Krasnopolsky, EMC/NCEP, 5200 Auth Rd., Camp Springs, MD 20746;
e-mail: kvladimir@sun1.wwb.noaa.gov

and variational retrievals may require some special, simplified versions of FMs.

2. FORWARD AND INVERSE PROBLEMS IN REMOTE SENSING

2.1 Forward Models

The above consideration shows that both standard and variational retrievals require some kind of conversion procedure, either a TF (retrieval algorithm) or a FM, to relate geophysical parameters to satellite measurements. The FM and TF are solutions of a remote sensing forward or inverse problem respectively. A generic remote sensing forward problem is symbolically represented by eq. (4) where $s \in \Re^n$ is a vector of satellite measurements (vector of BT's in the case of SSM/I) and $g \in \Re^m$ is a vector of geophysical (atmospheric and surface) parameters which influence the measurement. For example, after some simplification and extensive empirical parametrization a radiative transfer FM for SSM/I BTs ($s = T_{v,\pi}$) may be reduced to a closed algebraic version (e.g., Wentz, 1997),

$$T_{v,\pi} = \varepsilon_{v,\pi}\tau_v T_S + T_U^v + (1-\varepsilon_{v,\pi})\tau_v(\Omega_{v,\pi} T_D^v + \tau_v T_{BC}) \qquad (5)$$

where all terms in eq. (5) are empirical functions of wind speed, W, columnar water vapor, V, columnar liquid water, L, and sea surface temperature, T_s. Such physically-based or radiative transfer-based forward models use many empirical data for parametrization. For example, Wentz (1997) used 35,650 buoy-SSM/I matchups and 35,108 radiosonde-SSM/I matchups to fit more than 100 empirical parameters contained in different terms of eq. (5). Finally, this SSM/I FM (5) may be formally written as a system of algebraic equations,

$$T_{v,\pi} = F_{v,\pi}(X), \quad where\ X = \{W, V, L, T_S\} \qquad (6)$$

An alternative empirical approach can be applied to develop empirical forward models (or geophysical model function) based on empirical data. If a set of collocated in space and time satellite s and ground g observation - matchup data set $\{s, g\}$ - are collected or simulated, then an empirical FM can be developed based on this data set. Recently an empirical neural network model has been developed for five SSM/I BTs by Krasnopolsky (1997a) based on only about 3,500 matchups (see Section 4.1). This model can be formally described by eq. (6); however, the function F, in this case, is different. It is important to note that such an empirical model requires much less empirical data for development than the physically-based FM (5), it is more accurate (see Table 1 below) and much simpler (crucial for the direct assimilation) than the latter.

2.2 Retrieval Algorithms

A retrieval algorithm is a particular representation for a TF (1) and is also a solution of the inverse problem. Physically-based retrieval algorithm is an inversion of a physically-based forward model, therefore, it requires a physically-based FM as a necessary prerequisite, and, as a consequence, a large amount of empirical data for development (e.g., Wentz, 1997). An empirical algorithm does not require a FM to be developed; but a representative matchup data set is a prerequisite in this case. If we consider again the SSM/I as an example, most of SSM/I empirical wind speed algorithms (including the latest NN algorithm) have been developed using data sets of about 3,500 matchups (an order of magnitude less than for the physically-based algorithm by Wentz, 1997). For these empirical algorithms the resulting accuracies of retrievals are comparable or even better (for NN algorithm) than accuracies for the physically-based algorithm (see Table 2 below).

The inversion technique which is usually applied in physically-based retrieval algorithms to invert FM was described by Wentz (1997) for SSM/I retrieval algorithm. In this case, the TF, $\boldsymbol{f}$, (1) is not determined explicitly, it is only determined implicitly for each BT vector $\{T_{u,n}\}$. Symbolically, the retrieval algorithm can be written as

$$\boldsymbol{X} = \boldsymbol{f}(\boldsymbol{T}, T_s) \qquad (7)$$

It is important to emphasize that the algorithm (7), by definition, is a multi-parameter algorithm, since it retrieves a vector $\boldsymbol{X}$ of several geophysical parameters (W, V, and L) simultaneously. In addition to BTs, $\boldsymbol{T}$, this algorithm requires a SST value T_s as an input to produce retrievals.

Empirical algorithms are based on an approach which, from the beginning, assumes the existence of an explicit analytical representation for a TF, $\boldsymbol{f}$. Some mathematical model, $\boldsymbol{f}_{mod}$, is usually chosen (usually some regression) which contains a vector of empirical parameters $\boldsymbol{a} = \{a_1, a_2, \ldots\}$,

$$g_i = \boldsymbol{f}_{mod}(\boldsymbol{T}, \boldsymbol{a}) \qquad (8)$$

where these parameters are determined from an empirical matchup data set $\{g_i, \boldsymbol{T}\}$. The subscript i in g_i stresses the fact that most of empirical retrieval algorithms are single-parameter algorithms; they retrieve only wind speed (Goodberlet, 1989), or water vapor (Alishouse, 1990), or cloud liquid water (Weng and Grody, 1994), etc. Single-parameter algorithms have certain problems which are discussed below.

3. MATHEMATICAL TOOLS FOR DEVELOPING EMPIRICAL FORWARD MODELS AND RETRIEVAL ALGORITHMS

The above considerations show that both empirical forward models (4 and 5) and retrieval algorithms (1 and 8) can be considered as mappings which map a vector of sensor measurements, $\mathbf{s}$ (or $\mathbf{T}$) $\in \Re^n$, to a vector of geophysical parameters, $\mathbf{g}$ (or $\mathbf{X}$) $\in \Re^m$ (TFs, $\mathbf{f}$) or vice versa (FMs, $\mathbf{F}$). These mappings are built, using discrete sets of collocated vectors $\mathbf{s}$ and $\mathbf{g}$ (matchup data sets $\{s_i, g_i\}$). Single-parameter algorithms (8) may be considered as degenerate mappings where a vector is mapped onto a scalar (or a vector space onto a line).

The linear regression (LR) is the most attractive tool for developing empirical algorithms. It is simple; it has a well developed theoretical basis which enables a user to perform various statistical estimates. In the case of the LR, a linear model is built for FM or TF. For example for (8) we have:

$$g_i = LR(\mathbf{T}, \mathbf{a}) = \sum_j a_j T_j$$

here LR means linear regression and $\mathbf{a}$ is a vector of unknown parameters (regression coefficients). The problem with the LR is that it works with high accuracy in a broad range of the variability of arguments only if the problem and the function which it represents (FM or TF), is linear. If the problem is nonlinear, the LR can give only local approximation, or, if it is applied globally, this approximation has poor accuracy.

Because, in general, forward models and TFs are nonlinear functions of their arguments, nonlinear regressions (NRs) are better suited for modeling forward models and TFs. NR may be applied in many different ways. For example, $\mathbf{f}_{mod}$ in (8) can be chosen as a complicated NR function:

$$g_i = \mathbf{f}_{NR}(\mathbf{T}, \mathbf{a}) \quad (9)$$

on the other hand, $\mathbf{f}_{mod}$ can be introduced as an expansion in a set of nonlinear functions $\{\varphi_j\}$:

$$g_i = \sum_j a_j \varphi_j(T) \quad (10)$$

In either case, if we use NR (9) or (10), we need to specify in advance a particular type of nonlinear function $\mathbf{f}_{NR}$, or φ_j which we use. In other words, we need to introduce in advance a particular kind of nonlinearity, which we use to approximate the FM or TF under consideration. This may not always be possible, because we may not know in advance what kind of nonlinear behavior a particular FM or TF demonstrates, or this nonlinear behavior may be different in different regions of the FM's or TF's domain. If an improper NR is chosen (by chance), it may represent a nonlinear FM or TF less accurate than a LR.

In the situation described above, where we do know that the TF or FM is nonlinear but do not know what kind of nonlinearity to expect, we need a flexible, self-adjustable approach that can accommodate various types of nonlinear behavior and represent a broad class of nonlinear mappings. Neural networks (NNs) are well suited for a very broad class of nonlinear approximations and mappings. It has been shown (e.g., Chen and Chen, 1995; Hornik, 1991; Funahashi, 1989; Gybenko, 1989) that a NN with one hidden layer can approximate any continuous mapping defined on compact sets in $\Re^n$. Thus, any problem which can be mathematically reduced to a nonlinear mapping like (1), (4), (6), (8), etc. can be solved using a NN with one hidden layer.

Here several main properties of NNs are presented which make them a very suitable generic tool for nonlinear mapping (and, therefore, for algorithm development). Some of these properties have been illustrated above, others are described in the literature:

- NNs are able to model complicated nonlinear input/output relationships (any continuous nonlinear mapping).
- NNs are robust with respect to random noise and sensitive to systematic, regular signals (e.g., Kerlirzin and Réfrégier, 1995).
- NNs are fault-tolerant. An output value is created using all weights and biases so that an error in one of them usually causes only minor change in the output value(Cheng, 1996).
- NNs are well-suited for parallel processing and hardware implementations (Cheng, 1996) (all neurons in the same layer are completely independent and can be evaluated simultaneously).
- While training the NN is sometimes time consuming, its application is not. After the training is finished (it is usually performed only once), each application of the trained NN is practically instantaneous (several tens of floating point additions and multiplications -- microseconds on modern computers).

4. NEURAL NETWORKS FOR SSM/I DATA.

In previous sections we discussed theoretical possibilities and premises for using NNs for modeling TFs and FMs. In this section we illustrate these theoretical considerations using applications of the NN approach to the SSM/I forward and retrieval problems. Many different retrieval algorithms and several forward models have been developed for SSM/I sensor; and several different databases are available for the algorithm development and validation. For a detailed discussion of the database used here see Krasnopolsky et al., (1996), and Krasnopolsky, (1997a).

4.1 NN empirical forward model for SSM/I.

The empirical SSM/I FM represents the relationship between a vector of geophysical parameters ***X*** and a vector of satellite BTs ***T***, where ***T*** = *{T19V, T19H, T22V, T37V, T37H}*, ***X*** = *{W, V, L,* T_s (or *SST)}*. Four geophysical parameters were included in ***X*** (wind speed, *W*, columnar water vapor, *V*, columnar liquid water, *L*, and *SST*) which are the main parameters determining satellite BTs, and which are used as inputs in the physically based FMs of Petty and Katsaros (1994) and Wentz (1997) (see Table 1). The NN, OMBFM1, which implements this FM has 4 inputs, *{W, V, L, SST}*, one hidden layer with 12 neurons, 5 standard BT outputs *{T19V, T19H, T22V, T37V, T37H}*, and 20 auxiliary outputs which produce derivatives of the outputs with respect to the inputs, or $\partial T_i / \partial x_j$. These derivatives constitute the Jacobian matrix ***K***[***X***] = $\{\partial T_i / \partial x_j\}$ which emerges in the process of direct assimilation of the SSM/IBTs when the gradient of the SSM/I contribution to the cost function (3), χ_s, is calculated.

Fig. 1 shows the OMBFM1 architecture. Since these auxiliary outputs (Jacobian matrix ***K***) are not independent, we did not include them in the error function during the training, hence, only the standard outputs ***T*** are involved in the training process. Including these additional outputs in the NN architecture simplifies the use of our NN FM for direct assimilation because, as we showed in Section 1, in the process of variational retrievals the FM and its derivatives have to be estimated about 3,000 times per satellite measurement.

The matchup database for the F11 SSM/I has been used for training (about 3,500 matchups) and testing (about 3,500 matchups) our forward model. Only matchups with $R \leq 15$ km and $t \leq 15$ min have been selected. The FM was trained on all matchups which correspond to clear + cloudy conditions in

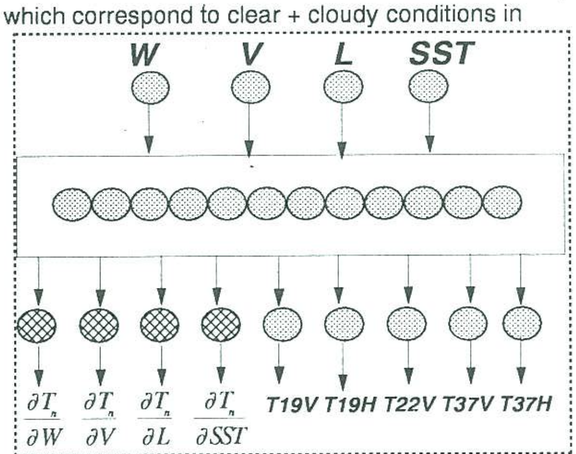

Fig. 1. NN forward model OMBFM1

Table 1. Comparison of physically based radiative transfer and empirical NN forward models for clear and clear+cloudy (in parentheses) conditions.

Author	Type	Inputs	BT RMS Error	
			V	H
Petty & Katsaros	Physically-based	*W, V, L, SST, Theta*[1], *P0*[2], *HWV*[3], *ZCLD*[4], *Ta*[5], *G*[6]	1.9 (2.3)	3.3 (4.3)
Wentz (1997)	Physically-based	*W, V, L, SST, Theta*[1]	2.3 (2.8)	3.4 (5.1)
Krasnopolsky	NN, empirical	*W, V, L, SST*	1.5 (1.7)	3.0 (3.4)

[1]*Theta* - incidence angle; [2] *P0* - surface presure
[3] *HWV* - vapor scale height; [4] *ZCLD* - cloud height
[5] *Ta* - effective surface temperature; [6] *G* - lapse rate

accordance with the retrieval flags introduced by Stogryn et al. (1994).

More than 6,000 matchups for the F10 instrument have been used for validation and comparison of the NN FM with physically-based forward models by Petty and Katsaros (1994) and Wentz (1997)[2]. For detailed training, testing and validation statistics see Krasnopolsky (1997a,b). Table 1 presents total statistics (RMS errors) for three FMs discussed here. RMS errors (in degrees Kelvin) are averaged over different frequencies for the vertical and horizontal polarizations separately for the F10 SSM/I data.

4.2 NN empirical SSM/I retrieval algorithms

About ten different SSM/I wind speed retrieval algorithms, both empirical and physically-based, have been developed using a large variety of approaches and methods. Here we perform a comparison of several of them in order to illustrate some properties of the different approaches mentioned in previous sections and some advantages of the NN approach.

The first global SSM/I wind speed retrieval algorithm (GSW) was developed by Goodberlet et al. (1989). This algorithm is a single-parameter algorithm (it retrieves only wind speed), and it is linear with respect to BTs (a multiple LR was used). This algorithm met specified accuracy criteria (2 m/s between 3 and 25 m/s) under rain-free and low moisture conditions. According to Goodberlet et al., however, this algorithm deteriorates rapidly in areas where rain and cloud cover occur. Wentz (1997) developed a physically-based approach to estimate surface wind speeds from the SSM/I. However, Wentz's approach requires some external inputs not available in the SSM/I data stream.

[2]The author coded both Wentz's FM and retrieval algorithm based on the detailed description published by Wentz (1997).

The first NN algorithm for SSM/I has been developed by Stogryn et al. (1994) for retrieving wind speed from the SSM/I BTs. This algorithm consists of two NNs, one of them performs retrievals under clear and another, under cloudy conditions. Krasnopolsky et al. (1994, 1995a) showed that a single NN (OMBNN1) with the same architecture can generate retrievals with the same accuracy as the two NNs developed by Stogryn et al. under both clear and cloudy conditions. This algorithm can be represented as:

$$W = f_{NN}(T) \qquad (11)$$

where W is the wind speed, and $\mathbf{T} = \{T19V, T22V, T37V, T37H\}$. Application of the OMBNN1 algorithm led to a significant improvement in wind speed retrieval accuracy for clear conditions. For higher moisture/cloudy conditions, the improvement was even far greater (25-30%) compared to the GSW algorithm. The increase in the areal coverage due to an improvements in accuracy was about 15% on average and higher in areas with higher meteorological activity.

First NN algorithms give very similar results because they have been developed using the same matchup database. This database, however, does not contain matchups with the wind speed higher than about 20 m/s and contains very few matchups with wind speeds higher than 15 m/s. These algorithms also are single-parameter algorithms, i.e. they retrieve only one parameter - wind speed, therefore they can not account for the variability of all related atmospheric (e.g., water vapor and liquid water) and surface (e.g., SST) parameters (especially important at higher wind speeds). This is why these NN algorithms demonstrate the same problem; they can not generate wind speeds higher then 18 - 19 m/s. The high wind speed performance has been improved in the OMBNN2 algorithm (Krasnopolsky et al., 1995b) by introducing new methods of NN training which enhance learning the high wind speed behavior and by using a bias correction. The OMBNN2 algorithm performs better than OMBNN1 for wind speeds higher than 15 m/s; however, it still can not generate wind speeds higher than 19 - 20 m/s without a bias correction because the same training set was used. It is also a single-parameter algorithm and is sensitive to the the variability of related atmospheric and surface parameters at higher wind speeds.

The next generation NN algorithm - a multi-parameter NN algorithm (OMBNN3; Krasnopolsky et al., 1996) solved the high wind speed problem through three main advances. First, a new buoy/SSM/I matchup database, created by NRL and enriched by high latitude, high wind speed events (up to 26 m/s) from European OWS MIKE and LIMA, was used for the development of this algorithm. Second, the method of NN training which enhances learning the high wind speed behavior was used. Third, the variability of the primary related atmospheric and surface parameters was taken into account: wind speed, columnar water vapor, columnar liquid water, and SST are retrieved simultaneously. In this case, the relation (11) is modified:

$$\mathbf{X} = f_{NN}(T) \qquad (12)$$

where $\mathbf{X} = \{W, V, L, SST\}$ is now a vector, and W is the wind speed, V - columnar water vapor, L - columnar liquid water, and SST - sea surface temperature. The OMBNN3 algorithm uses five SSM/I channels: 19 GHz and 37 GHz (horizontal and vertical polarization) and 22 GHz (vertical polarization). It does not use any additional inputs. SST is an output here rather than additional input as in Wentz algorithms.

Fig. 2 illustrates the evolution of our NN algorithms from OMBNN1 to OMBNN3. Table 2 shows a condensed comparison of the GSW, Wentz, and OMBNN3 algorithms for more than 15,000 matchups from three different SSM/I instruments F08, F10, and F11.

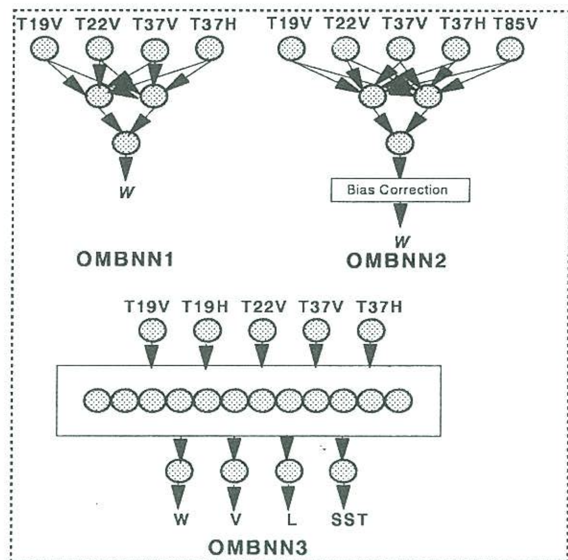

Fig.2 Evolution of the NN architecture from OMBNN1 to OMBNN3

Table 2. Biases and RMS errors (in m/s) for different SSM/I wind speed algorithms for clear and clear+cloudy (in parentheses) conditions.

Algorithm	Bias	RMS Errors
Linear Regression[1]	-0.2 (-0.5)	1.8(2.2)
Physically-Based[2]	0.5(0.2)	2.0(2.3)
OMBNN3[3]	-0.1(-0.2)	1.6(1.7)

[1]Goodberlet et al., 1989; [2]Wentz, 1997; [3]Krasnopolsky et al., 1996

The OMBNN3 algorithm demonstrates the best performance under both clear and clear+cloudy conditions.

In this section we have demonstrated that the NN FM and retrieval algorithm give results which are better than the results obtained with more sophisticated physically-based and empirical approaches. The NN FM simultaneously calculates the BTs and Jacobian matrix. It is much simpler and faster (only several tens of floating point additions and multiplications) than radiative transfer-based FMs. The OMBNN3 retrieval algorithm is also more accurate and robust than the physically-based and empirical algorithms developed for SSM/I retrievals (Krasnopolsky, 1997b).

4. CONCLUSIONS

In this work we discuss standard and variational (based on direct assimilation of satellite measurements) retrievals. Both standard and variational retrievals require a data convertor to convert satellite measurements into geophysical parameters or vice versa. Standard retrievals use a TF (solution of the inverse problem) and variational retrievals use a FM (solution of the forward problem) for this purpose. In many cases the TF and the FM can be represented as a nonlinear mapping. Because the NN technique is a generic technique for nonlinear mapping, it can be used advantageously for modeling TFs and FMs.

To illustrate advantages which one can get from applying the NN approach to the FM and TF development, we have presented a new NN-based empirical SSM/I FM called OMBFM1 and a new NN-based OMBNN3 transfer function (i.e., retrieval algorithm) for SSM/I retrievals. OMBFM1, given the wind speed, columnar water vapor, columnar liquid water, and *SST*, generates SSM/I BTs with acceptable accuracy. It is also significantly simpler than the physically-based FMs which is very important for variational retrievals where the FM is estimated several thousand times per satellite measurement.

The NN-based OMBNN3 transfer function (i.e., retrieval algorithm) for SSM/I retrieves the wind speed, the columnar water vapor, the columnar liquid water, and the *SST*. It demonstrates high retrieval accuracy overall, together with the ability to generate high wind speeds with acceptable accuracy. The OMBNN3 systematically outperforms all algorithms considered for all SSM/I instruments, under all weather conditions where retrievals are possible, and for all wind speeds. Some other applications of the NN approach to problems in the remote sensing can be found in a special issue of the International Journal of Remote Sensing (Neural Networks in Remote Sensing, 1997).

REFERENCES:

Alishouse, J.C., et al., Determination of oceanic total precipitable water from the SSM/I. *IEEE Trans. Geosci. Remote Sens., GE 23,* 811-816, 1990

Chen, C.H. (Editor in Ch
McGraw-Hill, New Yor
Chen, T., and H. Chen,
variables, nonlinear fu
neural networks," *Neu*
Eyre, J.R., and A.C. Lo
in numerical weather
1989
Funahashi, K., "On the
by neural networks," *N*
Goodberlet,M.A., C.T.Sv
ocean surface winds wit
94, 14574 - 14555.
Gybenko, G., "Approxi
in *Mathematics of Con*
314, 1989
Hornik, K., "Approxi
network,", *Neural Netw*
Kerlirzin, P., P. Réfrég
of multilayer perceptro
nonlinear networks,"
pp.560-571, 1995
Krasnopolsky V.M., L.(
a Single "All-Weather"
Winds from the Specia
OPC contribution No. 9
Krasnopolsky, V., L.C. B
a nonlinear transfer fu
from the special ser
100:11,033-11,045, 19
Krasnopolsky, V., W.H
wind speed retrievals
contribution No. 111, Env
1995b
Krasnopolsky, V., W.H
Function for SSM/I
Architecture". Technica
Krasnopolsky, V., "A
assimilation of SSM/I b
contribution No. 140, N
-"Neural networks for
Technical Note, OMB c
Lorenc, A.C., Analysis
R Meteorol. Soc., v. 1
Neural Networks in Re
Sensing, Special Issue
Parrish D.F., and J.C.
Spectral Statistical-Interp
1747-1763, 1992
Phalippou, L., Variatior
cloud liquid-water path
prediction., *Q. J. R. Me*
Petty G.W., and K.B. Kat
environment. Part II: A
slope distribution on e
Technol., 11, 617-628,
Prigent C., L. Phalippo
Observations during the
508, 1997
Stogryn A.P., C.T. Bu
retrievals from specia
networks" *JGR*, **90**, 98
Weng, F., and N.G. G
special sensor micro
25,535-25,551, 1994
Wentz, F.J., Measure
microwave radiometer
960-972, 1992.
Wentz F.J., "A well-c
microwave / imager", *JG*