FACTA UNIVERSITATIS
Series: **Architecture and Civil Engineering** Vol. 2, N° 3, 2001, pp. 185 - 191

ESTIMATION OF FAO PENMAN *C* FACTOR BY RBF NETWORKS

UDC 626.85 (045)

Slaviša Trajković[1], Branimir Todorović[2], Miomir Stanković[2]

[1] Faculty of Civil Engineering and Architecture, University of Niš,
Aleksandra Medvedeva 14, 18000 Niš, Serbia and Montenegro,
E-mail: slavisa@mail.gaf.ni.ac.yu
[2] Faculty of Occupational Safety, University of Niš, Čarnojevića 10a, 18000 Niš

Abstract. *An accurate estimation of the c factor is necessary in order to improve the validity of evapotranspiration estimation by the FAO (United Nations Food and Agriculture Organization) Penman method. The calculation of the c factor using the table interpolation or regression expressions can lead to a considerable error that is directly transferred to the estimated evapotranspiration. This paper reviews the application of RBF (Radial Basis Function) networks to estimate the FAO Penman c factor. The values of the c factors obtained by RBF networks were compared to the appropriate c values produced using regression expressions. It was shown that the RBF networks ensure a better agreement with table c values, thus improving the accuracy of the estimation of reference crop evapotranspiration. At the end of the paper, an example that demonstrates the simplicity of the use of RBF networks and the accuracy of the c factor estimation is presented.*

Key words: *Evapotranspiration, FAO-24 Penman, Artificial Neural Networks*

1. INTRODUCTION

The Penman method is used worldwide in evapotranspiration estimation. A frequently used version is (Doorenbos and Pruitt 1977):

$$ET_0 = c\,[WR_n + (1-W)0.27(1+0.01U_2)(e_a - e_d)] \quad (1)$$

where ET_0 = reference crop evapotranspiration (mm/d); c = adjustment factor; W = psychrometric weighting function; R_n = net radiation (mm/d); U_2 = mean wind speed at 2 m (km/d); e_a = saturation vapor pressure (millibars); and e_d = actual vapor pressure (millibars).

Received May 12, 2001

Tabular values of the *c* factor are given in Appendix II of Doorenbos and Pruitt (1977). The *c* factor is shown as a function of daily global solar radiation, R_S; maximum daily relative humidity, RH_{max}; mean daytime wind speed, U_d; and the ratio of daytime to nighttime wind speeds U_d / U_n.

The psychrometric weighting function is the weighting factor for the effect of radiation on reference crop evapotranspiration. A table defining W is provided in Table 4 of Doorenbos and Pruitt's paper. Net radiation R_n is the difference between all incoming and outgoing radiation and estimated as a function of the extraterrestrial radiation, R_a, and the maximum sunshine hours, *N*. Parameters R_a and *N* can be obtained from tables for specific latitude and months (Doorenbos and Pruitt, 1977); or it may be calculated using equations (Allen et al., 1989), (Jensen et al., 1990). Values of saturation vapor pressure e_a can be determined from Table 5 of Doorenbos and Pruitt's paper.

The values of parameters *W*, R_a, *N* and e_a can be easily obtained by table interpolation. Most often, only one interpolation is needed to obtain the accurate values of the appropriate parameter. The value of the *c* factor can also be obtained using table interpolation. However, it is necessary to make 15 interpolations in order to obtain that value, which requires the introduction of 45 different numbers into the calculation. Using this way of calculation with 15 interpolations, can lead to the possibility of making an error. Besides that, more time is needed to obtain *c* value (a few minutes).

The second approach that is used for estimating *c* values requires the use of regression expressions first introduced by Frevert et al. (1983) and later improved by Allen and Pruitt (1991). In spite of the improvements, the number of *c* factors with an error bigger than 4% is still a large one. It shows that it is necessary to develop a new approach of estimating the values of *c* factors. The aim of this paper is to develop new approach based on the RBF networks that would be simple to use, because it wouldn't demand from a user any background knowledge of Artificial Neural Networks (ANNs).

2. RBF NETWORKS

Artificial neural networks, as universal function approximators, are widely used for a variety of applications such as pattern recognition, control, identification and prediction of nonlinear dynamical systems. Most commonly applied learning algorithm is back propagation (Mason and Wang, 1990), and usually this algorithm is blamed for slow convergence speed, making it difficult to put applications to practice. However, the radial basis function network (RBF) proposed by Powell has a very fast convergence property, compared to multilayer perceptron (Moody and Darken, 1988), (Park and Sandberg, 1991). An arbitrary function can be approximated by the linear combination of locally tuned factorable basis functions. The property of locality is the main reason, why the RBF network can be learned much faster than the multy-layered perceptron. More detailed explanations about the RBF network architecture and network operation can be found in Fernando and Jayawardena (1998).

The RBF network has the input layer with N^I nodes, the hidden layer with N^H nodes and output layer with N^0 nodes. The internal units form a single layer of N^H receptive fields that can give the localized response function in the input space (Figure 1).

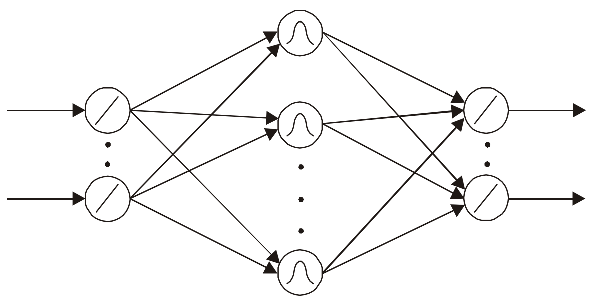

Fig. 1. A RBF network. Unit response functions are depicted graphically

The overall response function of the RBF network is:

$$\bar{x}^{O,\lambda} = \bar{\varphi}(\bar{x}^{I,\lambda}) = \mathbf{A} \cdot \bar{g}(\bar{x}^{I,\lambda}) \tag{2}$$

$$\bar{x}^{I,\lambda} = [x_1^{I,\lambda}, x_2^{I,\lambda}, ..., x_{N^I}^{I,\lambda}] \tag{3}$$

$$\bar{x}^{O,\lambda} = [x_1^{O,\lambda}, x_2^{O,\lambda}, ..., x_{N^O}^{O,\lambda}] \tag{4}$$

$$\bar{g} = [g_1(\bar{x}^{I,\lambda}), g_2(\bar{x}^{I,\lambda}), ..., g_{N^H}(\bar{x}^{I,\lambda})] \tag{5}$$

where $\bar{x}^{I,\lambda}$ is a real-valued vector in the input space; $\bar{x}^{O,\lambda}$ is a vector of output neurons activities; $\bar{g}$ is a vector of response functions of the i-th receptive field and $\mathbf{A} = [a_{ki}]$, $k = 1,...,n^O$, $i = 1,..., n^H$, are the output coefficients.

Several functions can be used as $g_i(\bar{x}^{I,\lambda})$, and in this paper is used Gaussian type radial basis function, which is given by:

$$g_i(\bar{x}^{I,\lambda}) = e^{-\sum_{j=1}^{n^I}\left(\frac{x_j^{I,\lambda} - m_{ij}}{\sigma_{ij}}\right)^2} \tag{6}$$

where m_{ij} and σ_{ij}, are the center and the width of radial basis function g_i for j-th input.

Task that the learning algorithm should perform can be formulated as follows. Given Λ input/output data and the specified model error $\varepsilon > 0$, obtain the optimal solution for network parameters a_{ki}, m_{ij}, σ_{ij}, $k = 1,...,n^O$, $i = 1,..., n^H$, $j = 1,..., n^I$, which satisfies the inequality:

$$\left(E = \frac{1}{2}\sum_{\lambda=1}^{\Lambda} \| \bar{x}^{O,\lambda} - \bar{x}^{*,\lambda} \|^2 \right) < \varepsilon \tag{7}$$

Parameter tuning process is based on the gradients $\partial E/\partial a_{ki}$, $\partial E/\partial m_{ij}$, $\partial E/\partial \sigma_{ij}$, $k = 1,...,n^O$, $i = 1,..., n^H$, $j = 1,..., n^I$, derivation:

$$E=\sum_{\lambda=1}^{\Lambda}E^{\lambda}\ ,\ E^{\lambda}=\frac{1}{2}\sum_{k=1}^{n^o}(x_k^{O,\lambda}-x_k^{*,\lambda})^2 \tag{8}$$

$$\frac{\partial E^{\lambda}}{\partial a_{ki}}=\frac{\partial E^{\lambda}}{\partial x_k^{O,\lambda}}\frac{\partial x_k^{O,\lambda}}{\partial a_{ki}}=(x_k^{O,\lambda}-x_k^{*,\lambda})\cdot g_i^{\lambda}(x^{I,\lambda}) \tag{9}$$

$$\frac{\partial E^{\lambda}}{\partial m_{ij}}=\frac{\partial E^{\lambda}}{\partial g_i^{\lambda}}\frac{\partial g_i^{\lambda}}{\partial m_{ij}}=\left(\sum_{k=1}^{n^o}\frac{\partial E^{\lambda}}{\partial x_k^{O,\lambda}}\frac{\partial x_k^{O,\lambda}}{\partial g_i^{\lambda}}\right)\frac{\partial g_i^{\lambda}}{\partial m_{ij}} \tag{10}$$

$$\frac{\partial E^{\lambda}}{\partial \sigma_{ij}}=\frac{\partial E^{\lambda}}{\partial g_i^{\lambda}}\frac{\partial g_i^{\lambda}}{\partial \sigma_{ij}}=\left(\sum_{k=1}^{n^o}\frac{\partial E^{\lambda}}{\partial x_k^{O,\lambda}}\frac{\partial x_k^{O,\lambda}}{\partial g_i^{\lambda}}\right)\frac{\partial g_i^{\lambda}}{\partial \sigma_{ij}} \tag{11}$$

$$\sum_{k=1}^{n^o}\frac{\partial E^{\lambda}}{\partial x_k^{O,\lambda}}\frac{\partial x_k^{O,\lambda}}{\partial g_i^{\lambda}}=\sum_{k=1}^{n^o}(x_k^{O,\lambda}-x_k^{*,\lambda})\cdot a_{ki} \tag{12}$$

$$\frac{\partial g_i^{\lambda}}{\partial m_{ij}}=2g_i^{\lambda}\frac{x_j^{I,\lambda}-m_{ij}}{\sigma_{ij}^2} \tag{13}$$

$$\frac{\partial g_i^{\lambda}}{\partial \sigma_{ij}}=2g_i^{\lambda}\frac{(x_j^{I,\lambda}-m_{ij})^2}{\sigma_{ij}^3} \tag{14}$$

For parameter tuning appropriate gradient methods can be used such as the steepest descent method, "momentum" method, conjugate gradient method, and quasi-Newton method.

Tuning of parameter p using gradient method with fixes step size is defined by the following iterative process:

$$p(n+1)=p(n)+\Delta p(n) \tag{15}$$

$$\Delta p(n)=-\eta\frac{\partial E(n)}{\partial p(n)} \tag{16}$$

An enhanced version of back propagation uses momentum term and flat regions elimination. The momentum term introduces the old parameter change as a parameter for the computation of the new weight change. The momentum term is used in this paper.

This avoids the oscillation problems common with the regular back propagation algorithm when the error surface has a very narrow minimum area. The new parameter update is computed by:

$$\Delta p(n+1)=-\eta\frac{\partial E(n)}{\partial p(n)}+\alpha\Delta p(n) \tag{17}$$

where α is the momentum, specifying the influence of the previous parameter change, and η is the learning step.

The effect of the momentum is that flat regions of the error surface are traversed relatively rapidly with a few big steps, while the step size is decreased, as the surface gets rougher. This adaptation of the step size increase learning speed significantly.

3. ESTIMATION OF FACTOR *C*

The RBF networks used for estimation of the factor *c* have the following structure. There are four neurons in the input layer. Their number is defined by the fact that the values of the *c* factor depend on four variables (RH_{min}, R_s, U_d, and U_d / U_n). The number of neurons in the hidden layer varies from 6 to 60. There is one neuron in the output layer of the RBF networks.

Samples (192 tabular values of factor *c*) are divided in to two groups. For the RBF network training, 168 randomly chosen training samples (no underlined values in Table 1) were used. All samples (192 tabular values of the *c* factor) are used for verification of RBF networks, obtained in a stage of training. Thus, the *c* values produced by RBF networks can be compared to the regression estimates and table values. Twenty-four table *c* values found in the verifying set only were used for controlling the ability of the networks to generalize the knowledge obtained during the training stage.

Table 1. FAO Penman *c* factor (Doorenbos and Pruitt 1977)

	SOLAR RADIATION (mm/d)											
U_d	RH_{max} = 30%				RH_{max} = 60%				RH_{max} = 90%			
(m/s) (1)	3 (2)	6 (3)	9 (4)	12 (5)	3 (6)	6 (7)	9 (8)	12 (9)	3 (10)	6 (11)	9 (12)	12 (13)
(a) U_d/U_n = 4												
0	0.86	0.90	1.00	1.00	0.96	0.98	1.05	1.05	1.02	1.06	1.10	1.10
3	0.79	0.84	0.92	0.97	0.92	1.00	1.11	1.19	0.99	1.10	1.27	1.32
6	0.68	0.77	0.87	0.93	0.85	0.96	1.11	1.19	0.94	1.10	1.26	1.33
9	0.55	0.65	0.78	0.90	0.76	0.88	1.02	1.14	0.88	1.01	1.16	1.27
(b) U_d/U_n = 3												
0	0.86	0.90	1.00	1.00	0.96	0.98	1.05	1.05	1.02	1.06	1.10	1.10
3	0.76	0.81	0.88	0.94	0.87	0.96	1.06	1.12	0.94	1.04	1.18	1.28
6	0.61	0.68	0.81	0.88	0.77	0.88	1.02	1.10	0.86	1.01	1.15	1.22
9	0.46	0.56	0.72	0.82	0.67	0.79	0.88	1.05	0.78	0.92	1.06	1.18
(c) U_d/U_n = 2												
0	0.86	0.90	1.00	1.00	0.96	0.98	1.05	1.05	1.02	1.06	1.10	1.10
3	0.69	0.76	0.85	0.92	0.83	0.91	0.99	1.05	0.89	0.98	1.10	1.14
6	0.53	0.61	0.74	0.84	0.70	0.80	0.94	1.02	0.79	0.92	1.05	1.12
9	0.37	0.48	0.65	0.76	0.59	0.70	0.84	0.95	0.71	0.81	0.96	1.06
(d) U_d/U_n = 1												
0	0.86	0.90	1.00	1.00	0.96	0.98	1.05	1.05	1.02	1.06	1.10	1.10
3	0.64	0.71	0.82	0.89	0.78	0.86	0.94	0.99	0.85	0.92	1.01	1.05
6	0.43	0.53	0.68	0.79	0.62	0.70	0.84	0.93	0.72	0.82	0.95	1.00
9	0.27	0.41	0.59	0.70	0.50	0.60	0.75	0.87	0.62	0.72	0.87	0.96

A network with ten neurons in the hidden layer, which gave the minimum error at the verifying stage, was chosen for use in the PROBA computer program. Table 2 shows the comparison of the c values produced by RBF networks or regression expressions with 192 tabular c values. In this table MARE denotes mean absolute relative error; MAXPRE the maximum positive relative error; MAXNRE the maximum negative relative error; NE the number of test samples with an error greater than 4% (NE>4%); DEV is the standard deviation of absolute relative error and r^2 is the coefficient of determination.

Table 2. Comparison of various methods used to calculate c factor

Model (1)	MARE (%) (2)	MAXPRE (%) (3)	MAXNRE (%) (4)	NE>4% (5)	DEV (%) (6)	r^2 (7)
Frevert et al. (1983)	3.686	26.16	13.04	58	3.744	0.955
Allen and Pruitt (1991)	2.927	32.41	13.26	36	3.981	0.979
RBF network	0.541	2.84	2.94	0	0.523	0.999

The mean absolute relative error of the RBF network is 0.54%, while the corresponding error of the regression expressions is 2.93% (Allen and Pruitt 1991), and 3.69% (Frevert et al. 1983). The RBF network has maximum error less than 3%, while in the regression expressions it is over 32%, and 26%, respectively. The number of samples with error greater than 4% in the regression models is 36, and 58 respectively, while in the RBF networks there is not a factor with such a large error.

4. Application

The following section includes examples for applying a new approach for factors estimation. The example data set is for average climatological variables at Beograd, Serbia and Montenegro during April, July and September from 1971 to 1975. The use of a trained RBF network is very simple and does not require any knowledge of ANN. This was achieved using the PROBA computer program, which requires a trained RBF network and a file with the input data (RH_{min}, R_s, U_d, and U_d / U_n). The agreement between the c values produced by trained RBF network and the table interpolation is great. Using the RBF network for the April, where variables $RH = 72.4\%$, $R_s = 6.34$ mm/d, $U_d = 3.14$ m/s and $U_d / U_n = 1.13$, c is equal to 0.886. The interpolated c value from Table 1 for the same data is 0.896 (difference of 1.1%). Applying the RBF network for the July, where variables $RH = 76.4$ %, $R_s = 8.74$ mm/d, $U_d = 2.22$ m/s and $U_d / U_n = 1.22$, c is equal to 0.991. The interpolated c value from Table 1 for the same data is 1.009 (difference of 1.7%). Using the RBF network for the September, where variables $RH = 84.2\%$, $R_s = 5.68$ mm/d, $U_d = 2.34$ m/s and $U_d / U_n = 1.05$, c is equal to 0.934. The interpolated c value from Table 1 for the same data is 0.931 (difference of 0.3%). RBF networks in comparison with table interpolation obtain the c values twenty times faster. The copy of PROBA computer program with a trained RBF network can be obtained from the first author.

5. Conclusions

The validity of evapotranspiration calculation by the FAO Penman method is increased with the accurate estimation of *c* factors. The determining of *c* values by table interpolation should be avoided because of its long procedure that can lead to a high error, which is directly transferred to the estimated evapotranspiration (see the expression (1)). The application of the regression expressions, in spite of the improvements done by Allen and Pruitt, does not always give satisfactory results. The comparative analysis showed that the RBF networks guarantees a more accurate estimation of *c* factors when compared to regression expressions. The use of the PROBA computer program with a trained RBF network is very simple and does not require any knowledge of ANNs.

References

1. Doorenbos, J., and Pruitt, W. O., *Crop water requirements*, FAO Irrigation and Drainage Paper No. 24, 2nd Ed., Food and Agricultural Organization of UN, Rome, Italy, 1977.
2. Allen, R. G., Jensen, M. E. , Wright, J. L. and Burman, R.D., *Operational Estimates of Reference Evapotranspiration*, Agronomy Journal, 81(4), 650-662, 1989.
3. M. E. Jensen, R. D. Burman, and R. G.Allen, *Evapotranspiration and Irrigation Water Requirements*, ASCE Manuals and Rep. on Engrg. Pract. No.70, ASCE, N.Y, 1990.
4. Frevert, D. K., Hill, R. W., and Braaten, B. C., *Estimation of FAO evapotranspiration coefficients*, J. Irrig. and Drain .Engrg., ASCE, 109(2), 265-270, 1983.
5. Allen, R. G., and Pruitt, W. O., *FAO-24 Reference Evapotranspiration Factors,* J. Irrig. and Drain. Engrg., ASCE, 117(5), 758-773, 1991.
6. Masson, E., and Wang, Y, *Introduction to computation and learning in artificial neural networks*, European Journal of Operational Research, 47, 1-28, 1990.
7. Moody, J., and Darken, C. J., *Learning with localized receptive fields*, Proceedings of the Connectionist Models Summer School, 133-143. 1988.
8. Park, J., and Sandberg, I. W., Universal approximation using radial basis function networks, Neural Computation, 1, 246-257, 1991.
9. Fernando, D. A., and Jayawardena, A. W., *Runoff Forecasting Using RBF Networks with OLS Algorithm*, Journal of Hydrologic Engrg., ASCE, 3 (3), 203-209, 1998.

ESTIMACIJA FAO-24 PENMAN *C* FAKTORA PRIMENOM RBF MREŽA

Slaviša Trajković, Branimir Todorović, Miomir Stanković

U ovom radu je prikazana estimacija FAO-24 Penman c faktora primenom RBF mreža. Vrednosti c faktora dobijene RBF mrežom su uporedjivane sa odgovarajućim vrednostima iz regresionih jednačina. Pokazano je da RBF mreža obezbedjuje bolje slaganje sa tabelarnim FAO-24 Penman c faktorima u poredjenju sa regresionim jednačinama. Na kraju rada je kroz praktičan primer pokazana sva jednostavnost korišćenja RBF mreže kao i pouzdanost proračuna FAO-24 c faktora.

Ključne reči: *Evapotranspiracija, FAO-24 Penman, Veštačke neuronske mreže*