Estimating Saturated Hydraulic Conductivity In Spatially Variable Fields Using Neural Network Ensembles

Kamban Parasuraman, Amin Elshorbagy,* and Bing Cheng Si

ABSTRACT

Modeling contaminant and water flow through soil requires accurate estimates of soil hydraulic properties in field scale. Although artificial neural networks (ANNs) based pedotransfer functions (PTFs) have been successfully adopted in modeling soil hydraulic properties at larger scales (national, continental, and intercontinental), the utility of ANNs in modeling saturated hydraulic conductivity (K_s) at a smaller (field) scale has rarely been reported. Hence, the objectives of this study are (i) to investigate the applicability of neural networks in estimating K_s at field scales, (ii) to compare the performance of the field-scale PTFs with the published neural networks program *Rosetta*, and (iii) to compare the performance of two different ensemble methods, namely Bagging and Boosting in estimating K_s. Datasets from two distinct sites are considered in the study. The performances of the models were evaluated when only sand, silt, and clay content (SSC) were used as inputs, and when SSC and bulk density ρ_b (SSC+ ρ_b) were used as inputs. For both datasets, the field scale models performed better than *Rosetta*. The comparison of field-scale ANN models employing bagging and boosting algorithms indicates that the neural network model employing the boosting algorithm results in better generalization by reducing both the bias and variance of the neural network models.

E STIMATION of the hydraulic properties of soils is of paramount importance for modeling contaminant and water flow through the vadose zone. Hydraulic properties also play an important role in partitioning the rainfall into runoff and soil moisture components. Soil hydraulic properties are usually measured in a laboratory using representative soil samples from the study area. Since the hydraulic properties exhibit large variations within a spatial domain, large numbers of soil samples are required to characterize the hydraulic properties of the study area. Laboratory estimates of hydraulic properties are complex and time-consuming; therefore, the interest in using PTFs to estimate the hydraulic property of the soil is increasing (Rawls and Brakensiek, 1983; Cosby et al., 1984; Saxton et al., 1986; Vereecken et al., 1990; van Genuchten et al., 1992; Leij et al., 2002).

Pedotransfer functions relate hydraulic properties to easily measurable or more widely available soil parameters (Bouma, 1989). A detailed review of different PTFs is given by Wösten et al. (2001). Pedotransfer function models include traditional regression models

(Wösten et al., 1995; Rawls et al., 1991) and ANNs (Schaap et al., 1998; Schaap and Bouten, 1996; Pachepsky et al., 1996; Minasny et al., 1999). A detailed review of ANNs and their application in predicting soil hydraulic properties can be found in Tamarai and Wösten (1999).

Schaap et al. (1998) showed that ANNs performed better than four published PTFs in estimating water retention data and six published PTFs in estimating the saturated hydraulic conductivity (K_s). The dataset used by Schaap et al. (1998) is derived from 4515 laboratory samples taken from 30 sources in the USA. Pachepsky et al. (1996) showed that the neural networks and regression models performed similarly in predicting the water retention parameters based on a dataset of 230 soil samples. Minasny and McBratney (2002) proposed a new objective function for neural network training, which predicted the parameters of the parametric model and optimized the PTF to match the observed and measured water contents. The study made use of 862 soil samples collected across Australia. Minasny and McBratney (2002) showed that their new objective function improved the performance of the neural network model when compared with the models employing traditional objective functions, in which the networks were optimized to fit the model parameters. Schaap et al. (2001) proposed a computer program, *Rosetta*, which implemented five hierarchical PTFs for the estimation of water retention and the saturated and unsaturated hydraulic conductivity. *Rosetta* is based on neural network analyses combined with the bootstrap method. The dataset used for constructing *Rosetta* was derived from soils in temperate to subtropical climates of North America and Europe. Most of the above discussed studies include a large number of samples obtained from a national scale, and it has been demonstrated that the ANNs are robust in predicting the hydraulic properties.

Traditionally, hydraulic properties are estimated from PTFs that were developed elsewhere (Tietje and Hennings, 1996; Tietje and Tapkenhinrichs, 1993). Hence, PTFs have been developed at various scales, including national (Nemes et al., 2003), continental (Wösten et al., 1999; Nemes et al., 2003), and intercontinental scales (Nemes et al., 2003). Nemes et al. (2003) showed that the PTFs developed at one scale were not suited for other scales. Moreover, they suggested that deriving PTFs from a small set of relevant data, when available, was more appropriate than using PTFs derived from a

K. Parasuraman and A. Elshorbagy, Centre for Advanced Numerical Simulation (CANSIM), Dep. of Civil & Geological Engineering, Univ. of Saskatchewan, Saskatoon, SK, Canada; B.C. Si, Dep. of Soil Science, Univ. of Saskatchewan, Saskatoon, SK, Canada. Received 30 Jan. 2006. *Corresponding author (Amin.Elshorbagy@usask.ca).

Abbreviations: ANNs, artificial neural networks; ρ_b, bulk density; FF-NN, feed-forward neural network; MR, mean residual; MRE, mean relative error; MSE, mean sum of squares of the network errors; PTF, pedotransfer function; RMSE, root mean squared error; SSC, sand, silt, and clay.

large but more general dataset. Also, according to Bastet et al. (1999), a particular PTF cannot be applied to the entire soil horizon; therefore, researchers should establish the validity of a PTF before adopting it. Pedotransfer functions developed at a large scale are best suited for global climate modeling, but might be of little use for modeling chemical transport and soil water balance on a farm field. The importance of the above considerations can be seen in the following example. Romano and Palladino (2002) examined the prediction of soil hydraulic properties along two linear transects based on soil physical properties and terrain information. Although efforts have been made to develop PTFs at large scales, little research has been conducted to evaluate the performance of ANNs in estimating saturated hydraulic conductivity (K_s) at field scale. Moreover, although the utility of the bagging algorithm in improving the generalization ability of ANNs models has been reported in various studies (Schaap et al., 1998, 2001; Nemes et al., 2003; Minasny et al., 2004), the ability of more versatile boosting algorithms (Schapire, 1990; Freund and Schapire, 1996) in improving the generalization ability of ANNs models in predicting K_s has not been investigated. Compared with bagging algorithm, boosting algorithm improves performance by producing a series of neural networks trained with a different distribution of the original training data.

The general objective of this study is to investigate the applicability of ANN-based PTFs at a field scale. The specific objectives include (1) determining the best combination of inputs in predicting K_s at field-scale, (2) comparing the performance of the field-scale PTF with the published neural networks program *Rosetta*, and (3) evaluating the relative performance of field-scale ANNs models employing bagging and boosting algorithms.

MATERIALS AND METHODS
Artificial Neural Networks

Feed-forward neural networks (FF-NNs) are the most widely adopted network architecture for the prediction and forecasting of geophysical variables (Maier and Dandy, 2000). Typically, FF-NNs consist of three layers: the input layer, hidden layer, and output layer. The number of nodes in the input layer corresponds to the number of inputs considered for modeling the output. The input layer is connected to the hidden layer with weights that determine the strength of the connections. The number of nodes in the hidden layer indicates the complexity of the problem being modeled. The hidden layer nodes consist of the activation function, which helps in nonlinearly transforming the inputs into an alternative space where the training samples are linearly separable (Brown and Harris, 1994). The most commonly used activation function is the sigmoidal transfer function as it is a bounded, monotonic, nondecreasing function that provides a graded, nonlinear response. The hidden layer is connected to the output layer. Detailed review of ANNs and their application in water sciences can be found in Maier and Dandy (2000) and in ASCE Task Committee on Application of Artificial Neural Networks in Hydrology (2000a, 2000b).

The structure of the three-layered FF-NN used in this study is shown in Fig. 1. The neural network model consists of '*j*'
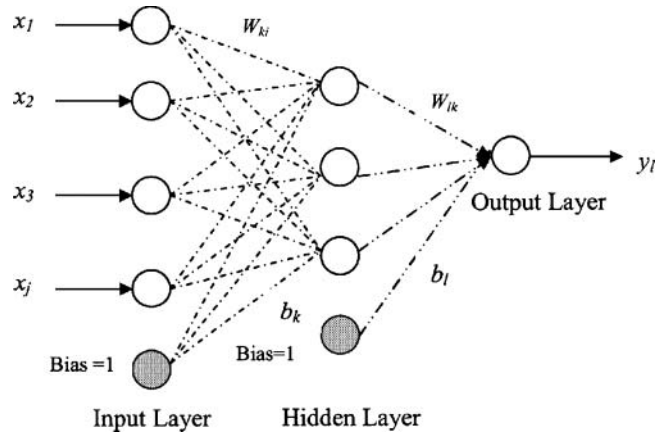
Fig. 1. Structure of the three-layered feed-forward neural network (FF-NN).

input neurons, '*k*' hidden neurons, and '*l*' output neurons. Symbolically, the ANN architecture shown in Fig. 1 can be represented as *ANN(j,k,l)*. The FF-NN adopted in this study makes use of the log-sigmoidal activation function in both the hidden layer and the output layer. In Fig. 1, W_{kj} represents the connection weight between the jth input neuron and k^{th} hidden neuron. Similarly, W_{lk} represents the connection weight between the kth hidden neuron and l^{th} output neuron. Parameters b_k and b_l represent the bias of the corresponding hidden and output layer neurons. The role of bias in a neuron is to displace the original functional domain by a magnitude equal to that of the bias and thereby translate the area of influence to its activation state. If x_j represents the input variables and y_l represents the output variables, then the inputs are transformed to output by the following equations (Haykin, 1999):

$$y_l = f_1 \left[\sum_{k=1}^{K} w_{lk} f_1 \left(\sum_{j=1}^{J} w_{kj}x_j + b_k \right) + b_l \right] \qquad [1]$$

$$f_1(p) = \frac{1}{(1 + e^{-p})} \qquad [2]$$

where $f_1(.)$ represents the log-sigmoidal activation function. The log-sigmoidal activation function helps in squashing the inputs between 0 and 1. One of the important issues in the development of a neural network model is the determination of optimal number of hidden neurons that can satisfactorily capture the nonlinear relationship existing between the input and the output variables. The number of neurons in the hidden layer is usually determined by the trial-and-error method with the objective of minimizing the cost function (ASCE Task Committee on Application of Artificial Neural Networks in Hydrology, 2000a). The typical cost function used in training FF-NNs involves minimizing the mean sum of squares of the network errors (MSE). In Eq. [3], y_i and y_i' represent the measured and computed counterparts, and n represents the number of training instances. A systematic search of different network configurations and user-adjustable parameters is performed to ascertain the optimal network architecture, with the objective of minimizing the cost function. The optimal network architecture is the one which results in the least cost function.

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - y_i')^2 \qquad [3]$$

The development of a neural network model demands two operations, namely, (i) training and (ii) testing. Training is a

process by which the connection weights between different layers and the bias values of the neural networks are optimized by minimizing the cost function. Since *Rosetta* uses Levenberg–Marquardt algorithm (Demuth and Beale, 2000), for a rational comparison of the proposed field-scale models with *Rosetta*, the same algorithm is adopted in this study to determine the optimal combination of connection weights and biases of the field-scale models. Once trained, the neural network model can be tested on an independent dataset that has not been used during the training process. More information on the Levenberg–Marquardt algorithm can be found elsewhere (Haykin, 1999).

One of the important properties of any neural network model is its generalization ability; that is, the ability of the neural network model to accurately predict the data that are not used for training the model. Recent theoretical and empirical studies have shown that the generalization ability of the neural network model can be improved by combining several neural network models in redundant ensembles. Hence in this study, the ANNs model is coupled with "bagging" (Breiman, 1996) and "boosting" (Schapire, 1990) algorithms, where several redundant ensembles of ANNs, created based on a statistical resampling technique (Efron and Tibshirani, 1993), are combined together to generate a unique output.

Bagging

Bagging (Breiman, 1996) is an acronym for "bootstrap aggregation." Using bagging, various datasets are generated from multiple realizations of the training dataset and these datasets are trained using different neural network models. The outputs from each of the neural network models are combined together to give a unique output. Moreover, bootstrapping allows the generation of an uncertainty estimate for each predicted value, which in turn aids the evaluation of the reliability of the model. The following paragraph outlines the methodology for carrying out bagging.

Suppose the training dataset T consists of N instances $(x_1, y_1),...,(x_N, y_N)$, where x and y are input and output variables, respectively. It is desired to obtain B bootstrap datasets. As a first step, each instance in T is assigned a probability of $1/N$, and the training set for each of the bootstrap member T_B is generated by sampling with replacement N times from the original dataset T using the above probabilities. Hence each bootstrap dataset T_B may have many instances in T repeated several times, while other instances may be left out. Individual neural network models are then trained on each of T_B. Therefore for any given input vector, the bootstrap algorithm provides B different outputs. The bagging estimate is then calculated by finding the mean of B different model predictions and the bagging uncertainty is estimated by finding the standard deviation of the B different model predictions.

Boosting

Compared with bagging, boosting algorithms (Schapire, 1990; Freund and Schapire, 1996) achieve improved performance by producing a series of neural networks trained with a different distribution of the original training data. The algorithm trains the initial neural networks with the original dataset and the training datasets for successive neural network models are assembled based on the performance of the current neural network model. If predicted values obtained from the current neural network model differ significantly from their observed values, the observed values will have higher probability of being selected in successive neural network models. In this way, the network is focused on learning hard patterns,

thereby improving the performance of the neural network model. In this study, the boosting algorithm ADABoost.R2 proposed by Drucker (1999) is adopted. ADABoost.R2 is a variation of the adaptive boosting algorithm, ADABoost.R proposed by Freund and Schapire (1996). Drucker (1999) showed that, in most cases, the ADABoost.R2 algorithm performed better than bagging in terms of prediction error when applied to ANNs. The ADABoost.R2 algorithm (Drucker, 1999) is detailed below.

Assume that the training dataset T consists of N instances $(x_1, y_1),...,(x_N, y_N)$, where x and y are input and output variables, respectively. Initially each value in the dataset is assigned the same probability value so that each instance in the initial dataset has an equal chance of being sampled in the first training set; that is, sampling distribution, $D_t(i)$ at step $t = 1$, is equal to $1/N$, over all i, where $i = 1$ to N. Iterate the following, while the average loss L, defined below, is less than 0.5 or a preset number of networks (t) are constructed.

1. Populate the new training set $NewT_t$ from the original training dataset T using the distribution D_t.
2. Construct a new network k_t, and train it using $NewT_t$.
3. Calculate the maximum loss, L_{max}, between the actual value and the network output $k_t(x_i, y)$, over the initial training set T where:

$$L_{max} = \sup\left(|k_t(x_i,y) - y_i|\right), \text{ over all } i \qquad [4]$$

Where sup() represents the maximum value of a set.

4. Calculate the individual L_i, loss for each element in the training set:

$$L_i = 1 - \exp\left[-\frac{|k_t(x_i,y) - y_i|}{L_{max}}\right] \qquad [5]$$

5. Calculate the weighted average loss, $\overline{L}$:

$$\overline{L} = \sum_{i=1}^{N} L_i D_t(i) \qquad [6]$$

6. Set β_t

$$\beta_t = \frac{\overline{L}}{1 - \overline{L}} \qquad [7]$$

7. Update the distribution D_i:

$$D_{t+1}(i) = \frac{D_t(i)\beta_t^{(1-L_i)}}{Z_t} \qquad [8]$$

Where Z_t is a normalization factor chosen such that D_{t+1} is a distribution.

8. Increment t by 1

For any given input vector, the boosting algorithm provides B different outputs, similar to bagging. Hence the boosting estimate and boosting uncertainty are estimated by finding the mean and standard deviation of the B different model predictions. The main difference between the neural network models employing the bagging and boosting algorithm is as follows: in the boosting algorithm, the distribution of the training set changes adaptively based on the performance of the previously created network, while the bagging algorithm changes the distribution of the training set stochastically. Although the boosting algorithms have better generalization ability than the bagging algorithms, the latter algorithm has the advantage of training the ensembles independently, hence in

parallel. In this study, in-house codes for bagging and boosting algorithms were developed using MatLab (the Mathworks, Lowell, MA). The performances of the models were evaluated when only sand, silt, and clay contents (SSC) were used as inputs, and when SSC and bulk density ρ_b (SSC+ ρ_b) were used as inputs. Using bagging and boosting algorithms, several redundant ensembles of ANNs, were created based on a statistical resampling of inputs (SSC and SSC+ ρ_b), and are combined together to generate a unique output (K_s). For both bagging and boosting algorithms, the optimal ensemble size, B, was found to be 30, using the trial-and-error method. Also, the number of hidden neurons in the networks employing bagging and boosting algorithms was determined using the trial-and-error method. Results from trial-and-error analysis indicated that the predictability of neural networks did not improve significantly with use of more than two hidden neurons. Hence the optimal number of hidden neurons for neural network models employing both bagging and boosting algorithms was two. Herein, the neural network model using the bagging algorithm will be referred to as *Field(Bagging)* and that using boosting algorithm will be referred to as *Field(Boosting)*.

Performance Evaluation

The performances of the different models are evaluated based on (i) root mean square error (RMSE), (ii) mean relative error (MRE), and (iii) mean residual (MR). RMSE, MRE, and MR statistics are calculated using Eq. [9], [10], and [11] respectively, where n represents the number of instances presented to the model and y_i and y_i' represent measured and computed K_s, respectively.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - y_i')^2} \qquad [9]$$

$$MRE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - y_i'}{y_i}\right| \qquad [10]$$

$$MR = \frac{1}{n}\sum_{i=1}^{n}(y_i - y_i') \qquad [11]$$

Since logarithmic values of K_s are considered, the corresponding RMSE, MRE, and MR statistics are dimensionless. Each of the above performance statistics provides different information about the predictive ability of the models. The RMSE statistic indicates only the model's ability to predict away from the mean (Hsu et al., 1995). RMSE gives more weight to high K_s values because it involves square of the difference between observed and predicted values. The MRE provides an unbiased error estimate because it gives appropriate weight to all magnitudes of the predicted variable. The closer to one is the ratio of predicted to measured, the smaller the MRE. This aspect of relative error is found to give a more appropriate assessment and comparison of different models (Legates and McCabe, 1999). The MR is a measure of prediction bias, with a negative and positive value of MR indicating underprediction and overprediction, respectively.

The best model should be unbiased (MR = 0), have the smallest MRE and have smallest overall dispersion (RMSE). In addition, the uncertainty or standard deviation among realizations of predicted values using Bagging or Boosting should be small. The uncertainty estimate, unlike the RMSE, MRE, and MR statistics, indicates the reliability of the K_s estimates. When no independent hydraulic data are available, calculation of RMSE, MRE, and MR statistics is not possible because it requires the measured K_s value. However, uncertainty esti-

mates can still provide the measure of reliability for the K_s, predicted by model. In this study, we consider these criteria equally important. Hence, to access the overall performance of each model, a rank score technique (Pandey and Nguyen, 1999; Shu and Burn, 2004) is adopted. To calculate the rank score, the models are ranked from best to worst according to the performance indices. Supposing that there are p models under consideration, a score of 1 is assigned to the best model and p for the worst model. For each model, the scores for the different performance indices are summed to obtain the overall performance score R_o for the model. Supposing that there are q indices, then the overall rank scores are in the range $[q, pq]$. R_o is then normalized to obtain the normalized rank score R_n, using the following equation, where the normalized rank scores are in the range $[0, 1]$, and an R_n close to 1 represents a model with good performance.

$$R_n = \frac{pq - R_o}{pq - q} \qquad [12]$$

Site Description and Sampling

Case Study I: Smeaton

The Smeaton research site is located at Smeaton, SK, Canada (53° 40′ N and 104° 58′ W). The soil at the site is classified as Gleyic Luvisol with texture dominated by sandy loam developed from glacio-fluvial and fluvial-lacustrine sands and gravels. The topography of the site is gently undulating and the climate is classified as cold and sub-humid. The long-term annual temperature, rainfall, and potential evapotranspiration are 0.1°C, 393 mm, and 530 mm, respectively (Anderson and Ellis, 1976).

A north-south transect of 384-m length was established on a gently sloping land with a variable texture and organic C content (Si and Zeleke, 2005; Zeleke and Si, 2005). After preliminary observations, a 3-m sampling interval was marked along the transect and core samples were collected in September, 2003 using 54-mm-diam. by 60-mm-long aluminium rings. All the 128 cores were used to determine the sand, silt, and clay content; and the bulk density (ρ_b) of the soil. Hydrometer method (Gee and Bauder, 1986) was used to determine the particle-size distribution. Saturated hydraulic conductivity (K_s) of the undisturbed core samples was determined using the constant head method (Klute and Dirksen, 1986). Of the 128 samples, two were considered outliers and the remaining dataset (126 samples) is split into training and testing datasets.

The dataset is split in such a way that every third instance appears in the testing set and the remaining instances make up the training set. This data segregation is performed to account for the spatial variability of the soil properties within both the training and testing datasets. The statistics of the entire dataset, along with the dataset used for training and testing are given in Table 1. Before modeling K_s using neural networks, the values of K_s are logarithmically transformed to avoid bias toward high conductivities. While sand content and ρ_b showed the least variation, clay content showed the highest variation. Moreover the training and testing datasets have similar statistical properties (Table 1).

Case Study II: Alvena

The Alvena research site is located at Alvena, SK, Canada (52° 31′ N and 106° 01′ W). The dominant soil type is an Aridic Ustoll and the landscape is classified as hummocky. The long-term annual temperature, rainfall, and potential evapotranspiration are 2.2°C, 350 mm, and 624 mm, respectively (Si and

Table 1. Statistics of the entire dataset along with the dataset used for training and testing (Smeaton).

Variable	Units	Entire Dataset (N = 126)					Training (N = 84)					Testing (N = 42)				
		Min.	Max.	Avg.	SD	CV	Min.	Max.	Avg.	SD	CV	Min.	Max.	Avg.	SD	CV
Sand	g g^{-1} × 100	52.50	86.30	65.00	7.30	0.11	53.80	86.30	65.90	7.60	0.12	52.50	76.30	63.20	6.29	0.10
Silt	g g^{-1} × 100	13.30	42.50	30.30	6.30	0.21	13.30	42.30	29.50	6.40	0.22	20.00	42.50	31.70	5.73	0.18
Clay	g g^{-1} × 100	0.50	15.00	4.90	2.70	0.55	0.50	15.00	4.80	2.80	0.58	1.30	11.30	5.20	2.51	0.48
ρ_b	g cm^{-1}	1.10	1.52	1.32	0.09	0.07	1.13	1.52	1.32	0.09	0.07	1.10	1.46	1.31	0.09	0.07
$\log_{10}(K_s)$	log10 (cm d^{-1})	1.13	2.29	1.86	0.26	0.14	1.13	2.29	1.84	0.28	0.15	1.49	2.29	1.91	0.21	0.11

Farrell, 2004). Undisturbed soil samples are collected along transect of 612 m length with a variable texture. The sand, silt, and clay content, along with the bulk density, were determined from these soil samples. Similar to the previous case study, particle-size distribution was determined based on the hydrometer method, and K_s was determined using the constant head method. Of the 78 samples, the training and testing datasets were selected similarly to the previous case-study. The training and testing sets consists of 52 and 26 samples, respectively.

The statistics of the entire dataset, along with the dataset used for training and testing, are given in Table 2. While silt content ranged from 46 to 63%, clay content ranged from 20 to 41%. Compared with the previous case study, the $\log_{10}(K_s)$ for the Alvena site showed higher variability. The coefficient of variation (CV) of $\log_{10}(K_s)$ for Smeaton and Alvena dataset is 0.14 $\log_{10}$(cm d^{-1}) and 0.24 $\log_{10}$(cm d^{-1}), respectively. The statistics of training and testing dataset are similar (Table 2).

RESULTS AND DISCUSSION

For the Smeaton case-study, the performance statistics of different models, when only three (SSC) inputs were used and when four (SSC+ ρ_b) inputs were used in predicting K_s, are presented in Table 3. Since Wösten (1990) recommended that the use of indirect methods for estimating hydraulic properties should be accompanied by the uncertainty of the estimations, the average uncertainty of the predicted K_s during both training and testing is also reported in Table 3. Along with RMSE, MRE, and MR statistics, the uncertainty statistics are also considered in calculating the rank score. The rank score presented in Table 3 is evaluated based on the performance of different models during testing.

When SSC was used as inputs for training, the field-scale models performed better than *Rosetta* (Table 3). This is expected because *Rosetta* is trained outside the field-scale dataset. However, the field-scale models also outperformed *Rosetta* in the testing set. The RMSE, MRE, and MR statistics achieved by *Rosetta* model were 0.26, 0.11, and 0.15 respectively, with an uncertainty value of 0.16 $\log_{10}$(cm d^{-1}). The field-scale models resulted in relatively smaller MR values than that of the *Rosetta*, indicating that the field-scale models were less biased. Also, the field-scale models resulted in smaller uncertainty values, thereby imparting more confidence

to the predicted values. The field-scale models using different algorithms, the Field(Bagging) and Field(Boosting), resulted in similar MREs. Nevertheless the Field(Boosting) model performed relatively better in terms of RMSE and MR. This is consistent with the findings of Drucker (1999), who reported that the ADABoost.R2 algorithm had better generalization property than the bagging algorithm. The Field(Boosting) algorithm resulted in an RMSE, MRE, and MR of 0.22, 0.09, and 0.07, respectively. Also, a least uncertainty value of 0.04 $\log_{10}$(cm d^{-1}) was achieved by both Field(Boosting) and Field(Bagging) models.

Along with SSC, when ρ_b was also used as one of the inputs (SSC+ρ_b) in predicting K_s, in general the performance of all the models improved considerably in terms of RMSE, MRE, and MR. With RMSE, MRE, and MR statistics of 0.19, 0.08, and −0.07, *Rosetta* showed the maximum improvement when ρ_b was added as input (Table 3). The *Rosetta* model, which over-predicted (positive MR) K_s when SSC was used as inputs, resulted in underprediction (negative MR) when SSC+ρ_b were used as inputs. Also the uncertainty in *Rosetta* model estimates dropped from 0.16 $\log_{10}$(cm d^{-1}) to 0.12 $\log_{10}$(cm d^{-1}) when ρ_b was added. However, opposite results were obtained in the case of field-scale models. When ρ_b was added to the inputs, the estimated uncertainty increased from 0.04 $\log_{10}$(cm d^{-1}) to 0.11 $\log_{10}$(cm d^{-1}) in the case of Field(Bagging) and from 0.04 $\log_{10}$(cm d^{-1}) to 0.10 $\log_{10}$(cm d^{-1}) in the case of Field(Boosting). Comparing the performance of *Rosetta* with the field-scale models, *Rosetta* resulted in the least RMSE statistics. However, the uncertainty estimates are larger than the field-scale models. The Field(Boosting) model resulted in the least MR and uncertainty statistics of 0.04 and 0.10 $\log_{10}$(cm d^{-1}). In general the overall performance of different models, as measured by their rank scores, indicated that the Field (Boosting) model performed better than other models, when either SSC or SSC+ρ_b was used as inputs. Figures 2 and 3 show the scatter plots between the measured and computed K_s using different models when SSC and SSC+ρ_b were used as inputs. In general, Rosetta underpredicted K_s when SSC were used as inputs and

Table 2. Statistics of the entire dataset along with the dataset used for training and testing (Alvena).

Variable	Units	Entire Dataset (N = 78)					Training (N = 52)					Testing (N = 26)				
		Min.	Max.	Avg.	SD	CV	Min.	Max.	Avg.	SD	CV	Min.	Max.	Avg.	SD	CV
Sand	g g^{-1} × 100	9.00	24.00	16.00	4.04	0.25	9.00	24.00	16.00	4.06	0.25	10.00	24.00	16.00	4.09	0.26
Silt	g g^{-1} × 100	46.00	63.00	55.00	4.18	0.07	47.00	63.00	55.00	4.10	0.07	46.00	62.00	55.00	4.43	0.08
Clay	g g^{-1} × 100	20.00	41.00	29.00	4.99	0.18	20.00	41.00	29.00	5.21	0.18	23.00	39.00	29.00	4.62	0.16
ρ_b	g cm^{-1}	1.11	2.35	1.33	0.17	0.14	1.11	2.35	1.33	0.19	0.14	1.14	1.57	1.32	0.12	0.09
$\log_{10}(K_s)$	log10 (cm d^{-1})	0.74	2.90	1.88	0.13	0.23	0.90	2.90	1.91	0.44	0.23	0.74	2.62	1.82	0.49	0.27

Table 3. Performance statistics of different models on the Smeaton dataset.

| Model	Training				Testing				Rank score
	RMSE	MRE	MR	Uncertainty	RMSE	MRE	MR	Uncertainty	
				SCC					
Rosetta	0.27	0.12	0.06	0.15	0.26	0.11	0.15	0.16	0.00
Field (Bagging)	0.23	0.10	−0.02	0.04	0.23	0.09	0.08	0.04	0.75
Field (Boosting)	0.23	0.11	−0.03	0.04	0.22	0.09	0.07	0.04	1.00
				SCC + ρ_b					
Rosetta	0.29	0.14	−0.17	0.12	0.19	0.08	−0.07	0.12	0.63
Field (Bagging)	0.21	0.10	−0.01	0.11	0.20	0.08	0.07	0.11	0.63
Field (Boosting)	0.22	0.10	−0.05	0.10	0.20	0.08	0.04	0.10	0.88

overpredicted K_s when SSC+ρ_b were used as inputs. However, the prediction trends of both the field-scale models were similar for both the input conditions (Fig. 2 and 3).

For the Alvena case-study the performances of different models as measured by RMSE, MRE, MR, along with uncertainty estimates, when SSC or SSC+ρ_b are used as inputs, are presented in Table 4. As with the previous case study, the rank score calculated based on the performance of different models during testing is also presented in Table 4. When SSC alone was used as inputs, the field-scale models outperformed *Rosetta* during both training and testing (Table 4). The MR statistics indicated that the field-scale models were less biased than *Rosetta*. *Rosetta* overpredicted K_s, and the field-scale models underpredicted K_s. The uncertainty estimate was 0.10 $\log_{10}$(cm d^{-1}) for *Rosetta*, 0.09 $\log_{10}$(cm d^{-1}) for Field(Bagging) and 0.10 $\log_{10}$(cm d^{-1}) for Field(Boosting). The Field(Boosting) model performed better than the Field(Bagging) model in terms of MR. However, the Field(Bagging) model resulted in smaller uncertainty value than Field(Boosting) model. Based on the rank score, the performances of both the field-scale models are similar.

When SSC+ ρ_b were used in estimating K_s, the field-scale models again outperformed *Rosetta* (Table 4). *Rosetta* resulted in RMSE, MRE, and MR estimates of 0.86, 0.42, and 0.60, respectively. An uncertainty of 0.12 $\log_{10}$(cm d^{-1}) was achieved by the *Rosetta* model. The Field(Boosting) model performed better than the Field(Bagging) model and the Field(Boosting) model resulted in the maximum rank score (Table 4). This illustrates the superior performance of the Field(Boost-

ing) model in predicting K_s. Moreover it should be noted that the addition of ρ_b as one of the inputs resulted in deterioration of the field-scale models performance during testing, although there was significant improvement during training (Table 4). This illustrates that the generalization property of the field-scale models is affected when ρ_b is considered as one of the inputs. Nevertheless, the addition of ρ_b as one of the inputs to the *Rosetta* model improved its performance during testing in terms of RMSE and MR, but deteriorated its performance in terms of MRE and the uncertainty estimate. The reason for the poor performances in the field-scale and *Rosetta* models is that ρ_b is poorly correlated to K_s at the Alvena site ($R^2 = 0.01$). Figures 4 and 5 show the scatter plots between the measured and computed K_s using different models when SSC and SSC+ ρ_b were used as inputs. From Fig. 4 and 5, it can be seen that the Rosetta model performed poorly in predicting K_s. Nevertheless, the performance of the local models was relatively better.

In general, it is observed that the performance of the neural network models in estimating K_s for the Smeaton dataset was better (less prediction errors) than that of the Alvena dataset. This is because soil hydraulic property depends on soil texture and soil structure. Soil structure in a sandy soil is dominantly single grained (or sometimes referred to as structureless). Hence substantial difference in K_s, due to soil structure, is unlikely in sandy soils. Nevertheless, soil structure in clay loam soil can be blocky, which in addition to soil texture, can introduce substantial difference in K_s. Since the neural network models consider only the soil texture, the better performance of the neural network models in sandy soils is expected.
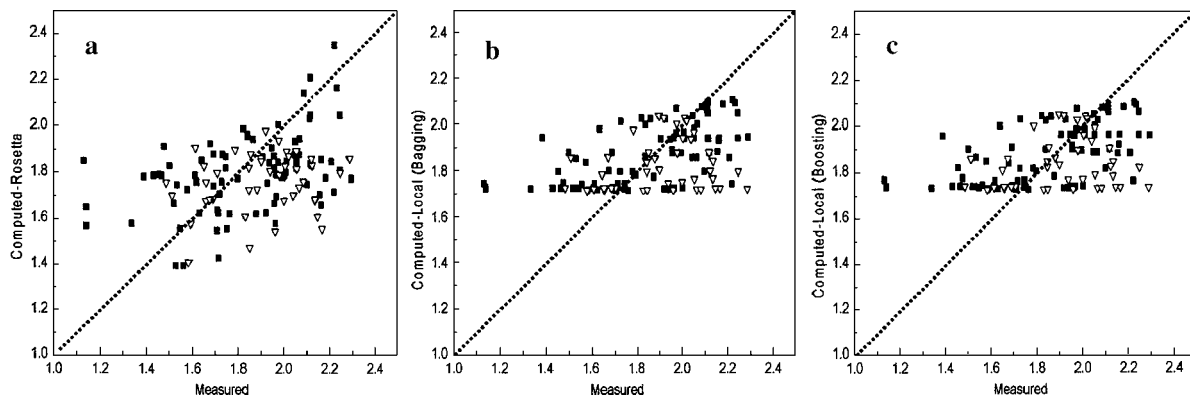
Fig. 2. Scatter plots between the measured and the computed K_s by (a) Rosetta; (b) Field(Bagging); and (c) Field(Boosting) for Smeaton with SSC as Inputs. The 'solid' points represent the training instances and the 'open triangular' points represent the testing instances.
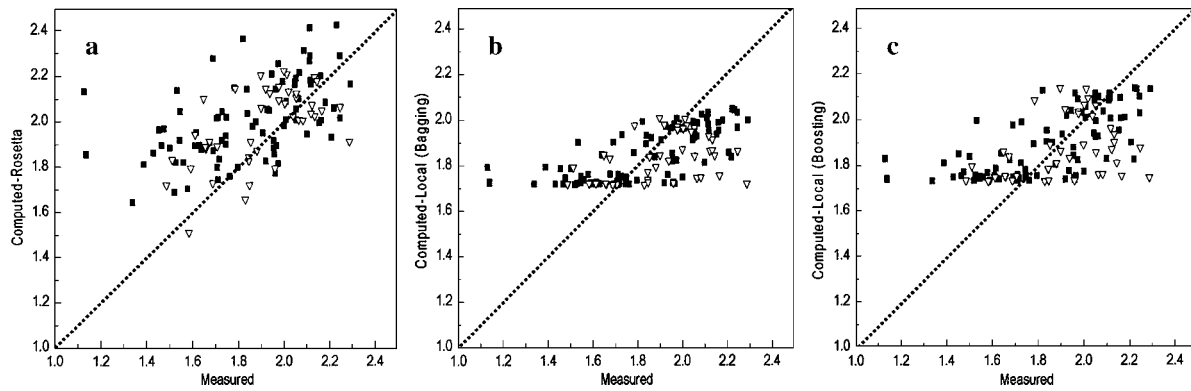
Fig. 3. Scatter plots between the measured and the computed K_s by (a) Rosetta; (b) Field(Bagging); and (c) Field(Boosting) for Smeaton with SSC and ρ_b as Inputs. The 'solid' points represent the training instances and the 'open triangular' points represent the testing instances.

Although the field-scale models outperformed *Rosetta* in both cases of the Smeaton (Table 3) and the Alvena (Table 4), the better performance of the field-scale models against *Rosetta* was more pronounced in the case of Alvena than in Smeaton. At the Smeaton site, the field-scale models had around 11% reduction in RMSE, 20% reduction in MRE, and 47% reduction in MR and 75% reduction in uncertainty when SSC were used as input. At the Alvena site, field-scale models had about 45% reduction in RMSE, 26% reduction in MRE, 74% reduction in MR, and 10% reduction in uncertainty when SSC were used as inputs. This may be due to the following reason: The soil type is sandy loam in Smeaton and silty-clay-loam in Alvena. Compared with the silty clay-loam soils, sandy soils are better represented in the training dataset of *Rosetta* (Schaap et al., 2001). Hence the performance of *Rosetta* was relatively better in the Smeaton case-study. The above finding is of particular significance because it reiterates the importance of the choice of proper training dataset. It can be concluded that the neural network model trained even on a small set of relevant data, when available, is better than training the neural network model with large but more general dataset. This finding is supported by Nemes et al. (2003). Moreover, the field-scale models are more parsimonious than *Rosetta* as the number of hidden neurons is six in *Rosetta*, and is two in the field-scale models. For both the case studies, the inclusion of ρ_b as one of the input parameters to the field-scale models, improved the performance of the models in terms of error estimates. However the uncertainty of the model predictions increased. Hence for the field-scale models, SSC is found to be the optimal combination of inputs.

As hypothesized, the boosting algorithm performs better than the bagging algorithm, which has been conventionally adopted in neural network modeling of soil hydraulic parameters. For both the case studies, the Field(Boosting) model resulted in a considerably less MR value than the Field(Bagging) model. This illustrates that the neural network model using boosting algorithm is less biased. This can be attributed to the ability of the boosting algorithm to focus and learn hard patterns, which in turn improves the performance of the neural network models. Unlike the bagging algorithm, which is largely a variance reduction method, the boosting algorithm is shown to reduce both bias and variance of the model. After each network in the ensemble is trained, the training samples with large errors have their weights increased while the training samples with small errors have their weights reduced for the purpose of training the next network in the ensemble. In this way, the boosting algorithm attempts to reduce the bias of the most recent network in the ensemble by focusing more on the training samples that have larger prediction errors.

In this study, we evaluated the performance of *Rosetta*, Field(Bagging), and Field(Boosting) models based on their ability in predicting the saturated hydraulic conductivity at field scales. More uncertainty analysis regarding the applicability of the neural networks predicted values in modeling the hydrological processes is beyond the scope of this study. Global models have wide applicability, but are found to perform poorly in field scale studies. Nevertheless, the most practical environmental and agricultural applications are at field scales. In this regard, the study is unique in that the performance of the ANNs models in predicting K_s at field scale

Table 4. Performance statistics of different models on the Alvena dataset.

| Model	Training				Testing				Rank score
	RMSE	MRE	MR	Uncertainty	RMSE	MRE	MR	Uncertainty	
				SSC					
Rosetta	0.92	0.40	0.81	0.11	0.87	0.38	0.72	0.10	0.13
Field (Bagging)	0.38	0.20	−0.09	0.10	0.48	0.28	−0.19	0.09	0.88
Field (Boosting)	0.38	0.20	−0.08	0.12	0.48	0.28	−0.17	0.10	0.88
				SSC + ρ_b					
Rosetta	0.96	0.38	0.70	0.13	0.86	0.42	0.60	0.12	0.00
Field (Bagging)	0.37	0.19	−0.13	0.12	0.50	0.29	−0.23	0.11	0.63
Field (Boosting)	0.36	0.18	−0.09	0.11	0.48	0.28	−0.19	0.11	1.00
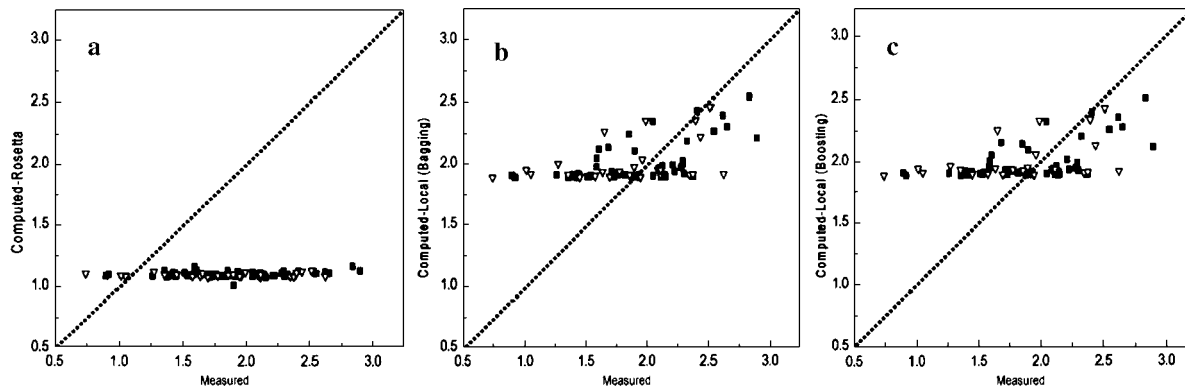
Fig. 4. Scatter plots between the measured and the computed K_s by (a) Rosetta; (b) Field(Bagging); and (c) Field(Boosting) for Alvena with SSC as Inputs. The 'solid' points represent the training instances and the 'open triangular' points represent the testing instances.

is explored. While one cannot extrapolate field-scale models to drastically different fields, field-scale models reduce the number of measurements required for that field. Also, in this study we illustrated the robustness of boosting algorithms in improving the generalization property of neural network models. Compared with the networks implementing the bagging algorithm, the neural network models implementing the boosting algorithm were shown to produce networks with less bias.

CONCLUSIONS

The study investigated the utility of neural network models in predicting the saturated hydraulic conductivity at field scale. Two different case-studies with different soil types were considered for the analysis. Two different ensemble neural network models, one using bagging algorithm and the other using boosting algorithm were developed and tested on the two case-studies. The performance of the field-scale artificial neural network models were compared with the published neural network program, *Rosetta*.

For both the case-studies, the field-scale neural network models performed better than the *Rosetta* model. This emphasizes that a neural network model trained even on a small set of relevant data, when available, is better than training a neural network model with a large but more general data set. For both the field-scale models, the inclusion of ρ_b as one of the inputs to the neural networks increased the uncertainty in the model predictions.

In contrast to most of the earlier studies that employed bagging algorithm to improve the performance of the neural network models in predicting the soil hydraulic properties, the study demonstrated the superior performance of the boosting algorithm based ensemble networks in modeling the saturated hydraulic conductivity at field scale. The Field(Boosting) model consistently resulted in less mean residual values than the Field (Bagging) model indicating the lower bias associated with the former model. Compared with the bagging algorithm, the boosting algorithm reduced both the bias and variance of the neural network models. The utility of the boosting algorithm in improving the performance of the neural network models with regards to modeling soil hydraulic parameters needs to be further explored on large scale databases.
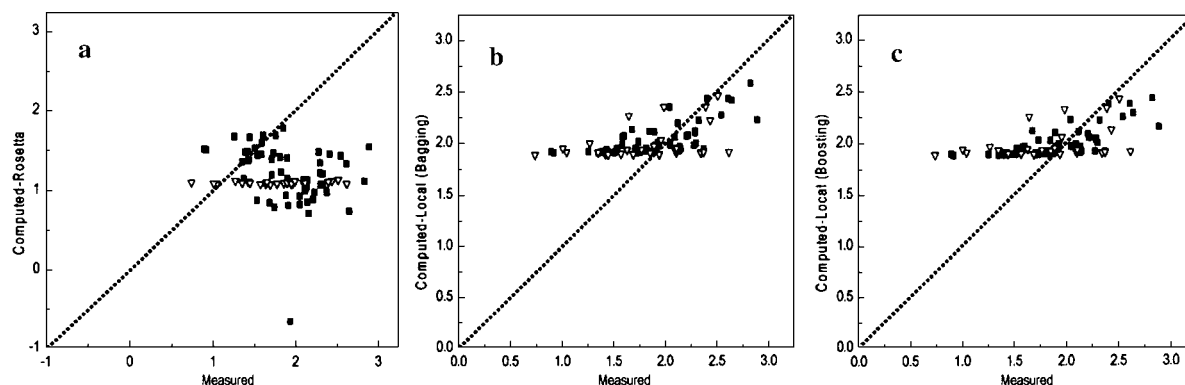
Fig. 5. Scatter plots between the measured and the computed K_s by (a) Rosetta; (b) Field(Bagging); and (c) Field(Boosting) for Alvena with SSC and ρ_b as Inputs. The 'solid' points represent the training instances and the 'open triangular' points represent the testing instances.

REFERENCES

Anderson, D.W., and J.G. Ellis. 1976. The soils of provincial forest reserves in the Prince Albert map area–73H Saskatchew.an. Saskatchewan Institute of Pedology, University of Saskatchewan, Saskatoon, Sask.

ASCE Task Committee on Application of Artificial Neural Networks in Hydrology. 2000a. Artificial neural networks in hydrology I: Preliminary concepts. J. Hydrol. Eng. 5:115–123.

ASCE Task Committee on Application of Artificial Neural Networks in Hydrology. 2000b. Artificial neural networks in hydrology II: Hydrologic application. J. Hydrol. Eng. 5:124–137.

Bastet, G., A. Bruand, M. Voltz, M. Bornand, and P. Quetin. 1999. Performance of available pedotransfer functions for predicting the water retention properties of French soils. p. 981–991. In M.Th. Van Genuchten et al (ed.) Characterization and measurement of the hydraulic properties of unsaturated porous media. Univ. of California, Riverside, CA.

Bouma, J. 1989. Using soil survey data for quantitative land evaluation. Adv. Soil Sci. 9:177–213.

Breiman, L. 1996. Bagging predictors. Mach. Learn. 24:123–140.

Brown, M., and C. Harris. 1994. Neurofuzzy adaptive modeling and control. Prentice Hall, New York.

Cosby, B.J., G.M. Hornberger, R.B. Clapp, and T.R. Ginn. 1984. A statistical exploration of the relationships of soil moisture characteristics to the physical properties of soils. Water Resour. Res. 20:682–690.

Demuth, H., and M. Beale. 2000. Neural network toolbox learning. For use with MATLAB. The Math Works Inc, MA.

Drucker, H. 1999. Boosting using neural networks. P. 51–78. In A.J.C. Sharkey (ed.) Combining artificial neural nets. Springer-Verlag, London.

Efron, B., and T.J. Tibshirani. 1993. An introduction to bootstrap. Chapman and Hall, New York.

Freund, Y., and R.E. Schapire. 1996. Experiments with a new boosting algorithm. p. 148–156. In L. Saitta (ed.) Proceedings of the Thirteenth International Conference on Machine Learning (ICML), Bari, Italy, 3–6 July 1996. Morgan Kaufmann, San Francisco, CA.

Gee, G.W., and J.W. Bauder. 1986. Particle-size analyses. p. 384–423. In A. Klute (ed.) Method of soil analyses. Part 1. Agron. Monogr. No. 9, ASA and SSSA, Madison, WI.

Haykin, S.S. 1999. Neural networks: A comprehensive foundation. Prentice Hall, NJ.

Hsu, K., V.H. Gupta, and S. Sorooshian. 1995. Artificial neural network modeling of the rainfall-runoff process. Water Resour. Res. 31:2517–2530.

Klute, A., and C. Dirksen. 1986. Hydraulic conductivity and diffusivity: Laboratory methods. p. 687–734. In A. Klute (ed.) Method of soil analyses. Part 1. Agron. Monogr. No. 9, ASA and SSSA, Madison, WI.

Legates, D.R., and G.J. McCabe, Jr. 1999. Evaluating the use of goodness-of-fit measures in hydrologic and hydroclimatic model validation. Water Resour. Res. 35:233–241.

Leij, F., M.G. Schaap, and L.M. Arya. 2002. Water retention and storage: Indirect methods. p. 1009–1045. In J.H. Dane and G.C. Topp (ed.) Methods of soil analysis. Part 4. SSSA Book Ser. No. 5. SSSA, Madison, WI.

Maier, H.R., and G.C. Dandy. 2000. Neural networks for the prediction and forecasting of water resources variables: A review of modeling issues and application. Environ. Model. Softw. 15:101–124.

Minasny, B., and A.B. McBratney. 2002. The neuro-m method for fitting neural network parametric pedotransfer functions. Soil Sci. Soc. Am. J. 66:352–361.

Minasny, B., J.W. Hopmans, T. Harter, S.O. Eching, A. Tuli, and M.A. Denton. 2004. Neural networks prediction of soil hydraulic functions for alluvial soils using multistep outflow method. Soil Sci. Soc. Am. J. 68:417–429.

Minasny, B., A.B. McBratney, and K.L. Bristow. 1999. Comparison of different approaches to the development of pedotransfer functions for water retention curves. Geoderma 93:225–253.

Nemes, A., M.G. Schaap, and J.H.M. Wösten. 2003. Functional evaluation of pedotransfer functions derived from different scales of data collection. Soil Sci. Soc. Am. J. 67:1093–1102.

Pachepsky, Y.A., D.J. Timlin, and G. Varallyay. 1996. Artificial neural networks to estimate soil water retention from easily measurable data. Soil Sci. Soc. Am. J. 60:727–773.

Pandey, G.R., and V.-T.-V. Nguyen. 1999. A comparative analysis of regression based methods in regional flood frequency analysis. J. Hydrol. 225:92–101.

Rawls, W.J., and D.L. Brakensiek. 1983. A procedure to predict Green and Ampt infiltration parameters. p. 102–112. In Adv. in Infiltration. ASAE, St. Joseph, MI.

Rawls, W.J., T.J. Gish, and D.L. Brakensiek. 1991. Estimating soil water retention from soil physical properties and characteristics. Adv. Soil Sci. 9:213–234.

Romano, N., and M. Palladino. 2002. Prediction of soil water retention using soil physical data and terrain attributes. J. Hydrol. 265:56–75.

Saxton, K.E., W.J. Rawls, J.S. Romberger, and R.I. Papendick. 1986. Estimating generalized soil-water characteristics from texture. Soil Sci. Soc. Am. J. 50:1031–1036.

Schaap, M.G., and W. Bouten. 1996. Modeling water retention curves of sandy soils using neural networks. Water Resour. Res. 32:3033–3040.

Schaap, M.G., F.L. Leij, and M.Th. Van Genuchten. 1998. Neural network analysis for hierarchical prediction of soil hydraulic properties. Soil Sci. Soc. Am. J. 62:847–855.

Schaap, M.G., F.L. Leij, and M.Th. Van Genuchten. 2001. Rosetta: A computer program for estimating soil hydraulic parameters with hierarchical pedotransfer functions. J. Hydrol. 251:163–176.

Schapire, R.E. 1990. The strength of weak learnability. Mach. Learn. 5:197–227.

Shu, C., and D.H. Burn. 2004. Artificial neural network ensembles and their application in pooled flood frequency analysis. Water Resour. Res. 40:W09301 doi:10.1029/2003WR002816.

Si, B.C., and R.E. Farrell. 2004. Scale-dependent relationship between wheat yield and topographic indices: A wavelet approach. Soil Sci. Soc. Am. J. 68:577–587.

Si, B.C. and T.B. Zeleke. 2005. Wavelet coherency to relate soil saturated hydraulic conductivity and physical properties. Water Resour. Res., 2005WR004118,41(11).

Tamarai, S., and J.H.M. Wösten. 1999. Using artificial neural networks to develop pedotransfer functions of soil hydraulic conductivity. p. 1251–1260. In M.Th. Van Genuchten et al (ed.) Characterization and measurement of the hydraulic properties of unsaturated porous media. Univ. of California, Riverside, CA.

Tietje, O., and M. Tapkenhinrichs. 1993. Evaluation of pedotransfer functions. Soil Sci. Soc. Am. J. 57:1088–1095.

Tietje, O., and V. Hennings. 1996. Accuracy of the saturated hydraulic conductivity prediction by pedo-transfer functions compared to the variability within FAO textural classes. Geoderma 69:71–84.

Van Genuchten, M.Th., F.J. Leij, and L.J. Lund. 1992. On estimating the hydraulic properties of unsaturated soils. p. 1–14. In M.Th. van Genuchten et al. (ed.) Indirect methods for estimating the hydraulic properties of unsaturated soils. Proc. Int. Workshop. Riverside, CA. 11–13 Oct. 1989. Univ. of California, Riverside.

Vereecken, H., J. Maes, and J. Feyen. 1990. Estimating unsaturated hydraulic conductivity from easily measured soil properties. Soil Sci. 149:1–12.

Wösten, J.H.M. 1990. Use of soil survey data to improve simulation of water movement in soils. Ph. D. thesis. Univ. of Wageningen, the Netherlands.

Wösten, J.H.M., P.A. Finke, and M.J.W. Jansen. 1995. Comparison of class and continuous pedotransfer functions to generate soil hydraulic characteristics. Geoderma 66:227–237.

Wösten, J.H.M., A. Lilly, A. Nemes, and C. Le Bas. 1999. Development and use of a database of hydraulic properties of European soils. Geoderma 90:169–185.

Wösten, J.H.M., Y.A. Pachepsky, and W.J. Rawls. 2001. Pedotransfer functions: Bridging gap between available basic soil data and missing soil hydraulic characteristics. J. Hydrol. 251:123–150.

Zeleke, T.B., and B.C. Si. 2005. Scaling relationships between saturated hydraulic conductivity and soil physical properties. Soil Sci. Soc. Am. J. 69:1691–1702.