Flow prediction in vegetative channel using hybrid artificial neural network approach

Bimlesh Kumar

ABSTRACT

Channel flow–vegetation interaction has been extensively studied in the past few decades and many equations have been developed which essentially differ from each other in derivation and form. As the process is extremely complex, getting deterministic or analytical forms of process phenomena are too difficult. A hybrid neural network model (combining genetic algorithm with neural network), which is particularly useful in modeling processes about which adequate knowledge of the physics is limited, is presented here as a complementary tool to model channel flow–vegetation interactions in submerged vegetation conditions. The prediction capability of the model has been found to be satisfactory. The input significance of the different parameters has been analyzed in the present work in order to find out the influence of these parameters on channel flow velocity.

Key words | flow velocity, genetic algorithm, input significance, sensitivity analysis, vegetation density, vegetation height

Bimlesh Kumar
Department of Civil Engineering,
Indian Institute of Technology Guwahati,
Guwahati-781039,
India
E-mail: *bimk@iitg.ernet.in*

INTRODUCTION

Estimating the flow resistance of vegetation is of great importance in river management, since it may have a significant effect on the conveyance of the channel (Järvelä 2002). The presence of vegetation in a channel also leads to a change in the hydrodynamic characteristics of the channel (Nepf 1999; Shields & Cooper 2000; Millar 2002; Montgometry & Piegay 2003). Interactions between flow and vegetation are complex and depend on environmental factors and plant characteristics such as mean flow velocity, turbulence, channel morphology, water temperature, plant morphology, age and size, and the spatial distribution of plant patches (Yen 2002). Knowledge of flow–vegetation interactions is important for both engineering and ecological applications (Nikora *et al*. 2008). The literature is replete with experimental and numerical works on flow–vegetation interactions. Experiments in laboratory flumes have been carried out to quantify flow–vegetation interactions by various researchers (Wilson *et al*. 2003; Armanini *et al*. 2005). Green (2006) has done field measurements in natural vegetated fields. Also, detailed numerical simulations of flow through vegetation have been performed (Neary 2003; Choi & Kang 2004). Several empirical (Kouwen & Fathi-Moghadam 2000) and theoretical relations (Stone & Shen 2002; Stephan & Gutknecht 2002) have been proposed to describe flow–vegetation interactions. Huthoff *et al*. (2007) have ascertained that due to their simplicity, empirical equations have better field applicability than theoretical ones. However, empirical relations have the drawback that their applicability is limited to the range of conditions for which they were derived. On the other hand, theoretical descriptions are often complex. Besides, they may require poorly understood closure parameters and sometimes pose practical difficulties when gathering required input data (Huthoff *et al*. 2007). Galema (2009), based on data available in the literature, has compared different predictors of flow characteristics and concluded that no simple predictor exists for both conditions.

Data mining techniques can help in modeling such processes about which the level of available knowledge is too limited to put the relevant information in a mathematical framework (Bhattacharya *et al*. 2004). Data mining is presently being utilized in almost all branches of science as an alternative and complementary technique to the more

doi: 10.2166/hydro.2013.255

traditional physically based modeling systems. Use of artificial neural networks (ANNs) remains in the forefront of this complementary modeling practice. Neural network techniques have been used to study several hydrologic and hydraulic phenomena (Dolling & Varas 2002; Srinivasulu & Jain 2006; Kumar 2011; Wei 2013).

Although conventional ANN setups have been employed in several instances, there are certain difficulties that ANNs are prone to when used with the back propagation training algorithm. The selection of network architecture and its efficient training procedure are major obstacles for any ANN-based model. Due to the profile of the error hyper-surface, ANN generally misses out the global optima. More recent algorithms (genetic algorithm (GA), particle swarm optimization, etc.) from the paradigm of evolutionary computing have made it possible to overcome such limitations of training algorithms (Koza 1992; Giustolisi & Savic 2006). A GA could represent a valid alternative to ANNs to establish a relationship among the parameters (vegetation characteristics, channel characteristics, etc.) influencing the flow. GA, based on natural selection and genetic mechanism, is a global search that grabbles from one population of points to another. As the algorithm continuously samples the parameter space, the search is directed toward the area of the best solution (Goldberg 1989). GA has been applied to many problems in hydrology and water resources (Cheng *et al.* 2002; Wang *et al.* 2007; Nasseri *et al.* 2008).

The primary objective of the present work is to develop the flow prediction model of a submerged vegetative channel. Recent studies have provided a variety of methods for quantifying and interpreting the contributions of the variables in neural networks (Duh *et al.* 1998; Olden & Jackson 2002). Thus the methods developed for input significance testing through neural networks have been applied in the present work to find out variability of flow velocity with other different physical parameters.

FUNCTIONAL ANALYSIS OF THE FLOW–VEGETATION INTERACTION

The use of neural networks allows developing a model incorporating many complex variables of the systems to be built without requiring the explicit formulation of the possible relationships that may exist among variables. Neural network does function mapping of the input vectors of real values to the real output values. Thus, it needs a functional form of a physical system describing its dependent and independent variables. Yen (2002) has analyzed several equations in terms of their dependent and independent variables. The physical modeling of channel flow through vegetation is very complex. According to Lopez & Garcia (1998), it is a function of many variables: the fluid properties, flow properties, vegetation characteristics, and channel characteristics. A closer look (Table 1) at the existing equations shows that flow–vegetation interactions can be described by the following function:

$$f(u, h, k, i, m, C_d, D) = 0 \tag{1}$$

where u is the mean velocity, h is the flow depth, k is the height of the vegetation, i is the channel slope. D is the diameter of cylindrical vegetation and m is the number of cylinders per m^2 horizontal area. C_d is the non-dimensional drag coefficient. Equation (1) is for homogeneous vegetation of fixed height and diameter. Flow is assumed to be steady and uniform. Assuming a wide channel, sidewall effects can be neglected (Borovkov & Yurchuk 1994). The condition of fully submerged vegetation is $h > > 5k$. In this case, the vegetation does not block the velocity at the upper part of the water column (Galema 2009). The submerged condition also follows $5k > h > k$. In these conditions, the vegetation is relatively high in relation to the flow depth; as a

Table 1 | Available descriptors for flow velocity in vegetative channel (submerged vegetation)

Author(s)	Equation(s)
Stone & Shen (2002)	$u = \sqrt{\frac{2g}{C_d m D}}\sqrt{i}(1 - D\sqrt{m})\sqrt{\left(\frac{h}{k} - \frac{1}{4}\pi m D^2\right)\frac{h}{k}}$
Van Velzen *et al.* (2003)	$u = \sqrt{\frac{2g}{C_d m D}}\sqrt{i} + 18(h-k)^{3/2}\frac{\sqrt{i}}{k}\log\frac{12(h-k)}{1.6k^{0.7}}$
Baptist *et al.* (2006)	$u = \left(\sqrt{\frac{2g}{C_d m D}}\sqrt{i}\sqrt{k} + \frac{\sqrt{g}}{0.4}\ln\left(\frac{h}{k}\right)\sqrt{i}\right)\sqrt{h}$
Huthoff (2007)	$u = \sqrt{\frac{2g}{C_d m D}}\sqrt{i}\left(\sqrt{\frac{k}{h}} + \frac{h-k}{k}\left(\frac{(h-k)\sqrt{m}}{1 - D\sqrt{m}}\right)^{2/3}\right)$

consequence, the velocity profile changes a great deal over depth. In channels with sufficient density of vegetation, the effect of bottom roughness can also be neglected (Stone & Shen 2002). The present work aims at predicting the flow velocity of a submerged vegetated channel, so u can be made as dependent variable which depends on the various other factors as shown in Equation (1). Keeping this in view, Equation (1) can be written as:

$$u = f(h, k, i, m, C_d, D) \tag{2}$$

The objective here is to construct a neural network model that approximates an unknown input–output mapping of Equation (2) on the basis of given real observations. The goal, however, is not to provide an exact fit to the data but to develop a model that captures the underlying relationship so that it can be used to predict the output at some future observation of the input.

NEURAL NETWORK MODELING

In this paper, a hybrid algorithm using GA technique is proposed to optimize parameters of the ANN. This is done by considering the following steps: first, define the structure of the ANN; second, define the encoding of the interconnection weights; and third, optimize the interconnection weights of the ANN by GA. In this paper, the most popular models of neural networks (NNs) such as feedforward networks are considered. Supervised learning is one of the most effective weight training algorithms, whereby efforts are made to find an optimal set of connective weights for a NN according to some optimality criteria. One of the most popular supervised learning training algorithms for feedforward NNs is back propagation (BP). The BP is a gradient descent search algorithm. It is based on minimization of the total mean square error between the actual output and the desired output (Irie & Miyanki 1988). This error is used to guide the search of the BP algorithm in the weight space. However, the problem of the BP algorithm is that it is very often trapped in local minima and the learning and adaptation speed are very slow in searching for the global minimum of the search space. Some developed evolutionary algorithms, notably GA, have attracted great attention in the NN from many communities (Chen *et al.* 1999; Salajegheh & Gholizadeh 2005; Giustolisi & Simeone 2006; Singh *et al.* 2007). GAs are parallel stochastic optimization procedures which are good at exploring a large and complex space in an intelligent way to find values close to the global optimum (Goldberg 1989). As compared with BP, GA is more qualified for neural networks if only the requirement of a global searching is considered. A simple chromosome representation is used, which contains information about connections, weights, and biases of the multiple layer perceptron neural network (MPLNN). The parameter learning process, based on GA technique and BP algorithm, is a two-step learning process. In the first step, the initial parameters, such as weights of the NN are tuned by the GA. In the second step, the BP algorithm and the Levenberg–Marquardt method is introduced to train the NN to yield optimal values of weights of the NN. The block diagram of the proposed hybrid algorithm is depicted by Figure 1.

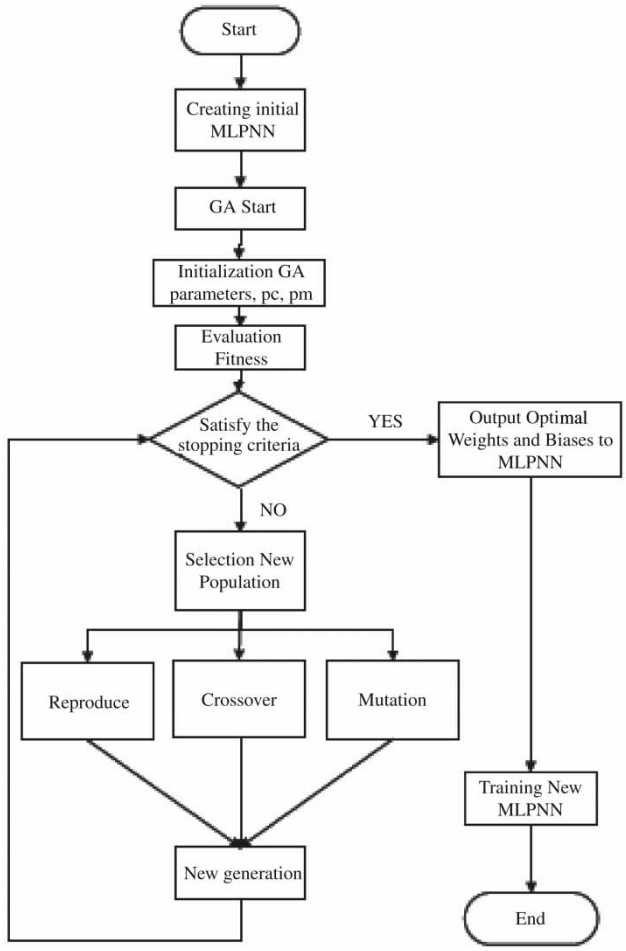

Figure 1 | Flowchart of the learning algorithm.

The procedure of the hybrid algorithm is presented as follows in the sections below.

Chromosome representation

An MLPNN can be represented by a directed graph, encoded on a chromosome with each parameter (weights and biases). All these parameters are memorized by a row vector $C = (c_i)$, $i = 1, 2 \ldots M$, where M is the number of all NN parameters. The chromosome can be written as:

$$C = [W_1, W_2] \tag{3}$$

where W_1 denotes the connective weight of link between the input layer and the first hidden layer and W_2 is the connective weight of link between the hidden layer and output layer.

Fitness function

The fitness function is dependent on problem and is used to evaluate the performance of each individual. The error signal of the output neuron j at iteration n (i.e., presentation of the nth training example) is defined by:

$$e_j(n) = d_j(n) - y_j(n), \quad \text{neuron } j \text{ is an output neuron} \tag{4}$$

The instantaneous value of the error energy for neuron j can also be defined as: $(1/2)e_j^2(n)$. Correspondingly, the value of $\xi(n)$ is obtained by summing $(1/2)e_j^2(n)$ over all neurons in the output layer (Haykin 1999).

$$\xi(n) = \frac{1}{2}\sum_{j=C} e_j^2(n) \tag{5}$$

where the set C includes all the neurons in the output layer of the network. For MLPNN it is the sum squared error. The fitness is defined as by summing $\xi(n)$ over all n with respect to the set size N, as shown by:

$$F = \sum_{n=1}^{N} \xi(n) \tag{6}$$

Here, the objective is to minimize $F(\cdot)$ subject to weights and biases.

Selection

The selection operator is to select individuals from the population for reproduction based on the relative fitness value of each individual. The extraction can be carried out in several ways. One of the most commonly used selection methods is the roulette wheel selection (Goldberg 1989), where individuals are extracted in probability following a Monte Carlo procedure. The extraction probability $p_r(Xi)$ of each individual X_i is proportional to its fitness $F(X\text{i})$ as a ratio to the average fitness of all the individuals. The offsprings are produced based on this selection.

Crossover

To apply the standard crossover operator the individuals of the population are randomly paired. Crossover takes two parents and performs an interpolation of the two parents. Each pair is then recombined, and the new individuals (offsprings) are formed by the interpolation of parents.

Mutation

After crossover, the new individuals are subjected to mutation. Mutation prevents the algorithm being trapped in a local minimum. A variable is selected with a certain probability and its value is modified by a random value. Here, we choose non-uniform mutation method. Non-uniform mutation changes one of the genes of the parent based on a non-uniform probability distribution.

Levenberg–Marquardt method

The equations for changing the weights (ΔW) during training in the Levenberg–Marquardt method are given as follows:

$$\Delta W = (J^T J + \mu I)^{-1} \quad J^T e \tag{7}$$

where J is the Jacobian matrix of the derivative of each error to each weight, μ is a scalar, and e is an error vector. The Levenberg–Marquardt algorithm performs very well and its efficiency is found to be of several orders above the

conventional back propagation with learning rate and momentum factor.

RESULTS AND DISCUSSION

The entire modeling and analysis has been done by the use of the neural network toolbox of MATLAB® software. Galema (2009) documented a comprehensive database of different types of vegetations with hydraulic properties from different sources. Table 2 shows the ranges of the observations used in the present work. The total number of observations used in the modeling is 449. The sample size of the data set is an important issue in neural networks. Neural networks typically require larger sample sizes than conventional statistical procedures for model building and validation. In general, the larger the sample is, the better is the chance for a neural network to adequately approximate the underlying complex patterns without suffering from the problem of overfitting/underfitting. For development of the ANN model, 60% of observations (independent of the source, data have been randomized before selecting training, test, and validation sets) have been assigned as training sets and 20% for testing and 20% for validation. All data sources are not cited in the references of this paper as they can be found in Galema (2009). It should be noted that, like all empirical models, ANNs perform best in interpolation rather than extrapolation (Masters 1993); consequently, the extreme values of the available data are included in the training set. Once data have been divided into their subsets, the input and output variables are preprocessed by scaling them between 0.0 and 1.0 to eliminate their dimension and to ensure that all variables receive equal attention during training. The architecture of the initial neural network (Figure 2) has been chosen based on maximum R^2 over complete observations with increasing number of nodes in the hidden layers (Appendix 1, available online at http://www.iwaponline.com/jh/016/055.pdf). The transfer function used in the model is log sigmoid and tan sigmoid at input to hidden and hidden to output layers, respectively. The number of iterations has been fixed at 1,000.

The results of neural modeling are shown in Figure 3. It can be clearly seen from Figure 3 that the linear coefficient of correlation is very high between observed data and values predicted through neural nets and it is 0.993, 0.984, and 0.988 in training, validation, and testing, respectively. Overall, the linear coefficient of correlation is 0.982 as shown in Figure 3. This shows the learning and generalization performance of the network is good. The non-linear and non-parametric nature of the neural network model is desirable for field applications, it also brings about more opportunities to go wrong in the modeling and application process. Thus, it is worthwhile noting that predictions from ANN models are better when used for ranges of input variables similar to those utilized in model training. Two standard training algorithms – the Levenberg–Marquardt (LM) optimization and the BFGS (Broyden, Fletcher, Goldfarb, and Shanno) quasi-Newton method along with multiple regression analysis have been used to compare the GA-ANN results. The performance of gradient-based methods (ANN-LM and ANN-BFGS) was found to be comparable with those of the GA-ANN algorithm. The model equation generated by multiple regression analysis is given in Appendix 2 (available online at http://www.iwaponline.com/jh/016/055.pdf). However, the results of training the algorithm with GA-ANN gave superior predictive models, as reflected in Table 3.

In the present work, formulas listed in Table 1 have also been tested to quantify their prediction capability. Performance analysis of these formulas is shown in Figure 4. As shown in Figure 4, Huthoff's (2007) formula predicts better than other formulas with R^2 around 0.67. The functional form of the empirical formulas does not hold true for all the ranges, for example, Huthoff's (2007) formula works only when h is greater than k. ANN is free from such functional form and will work on all the ranges used in deriving the model.

In the neural network, the connection weights between neurons are the linkages between the input and the output of the network, and therefore are the link between the problem and the solution (Olden & Jackson 2002). Garson's algorithm or 'weights' method includes partitioning the connection weights to determine the relative importance of the various inputs. In the present work, connection weights (Olden & Jackson 2002), Garson's algorithm (Garson 1991) and partial derivative method (Ng *et al.* 2004) have been implemented to know the importance of the input variables on output. Connection weights calculate the product of the raw input–hidden and hidden–output connection weights between each input neuron and output neuron and sum

Table 2 | Summary of hydraulic and vegetation characteristics of the data

		D (m)		*m* (m^{-2})		*k* (m)		C_d		*h* (m)		*i*		*u*	
Source	**No. of observations**	**Min value**	**Max value**	**Min value**	**Max value**	**Min value**	**Max value**	**Min value**	**Max value**	**Min value**	**Max value**	**Min value**	**Max value**	**Min value**	**Max value**
Einstein & Banks (1950)	20	0.0064	0.0064	3	108		0.038		1.4	0.073	0.108	0.00774	0.01024	0.8	1.2
Tsujimoto & Kitamura (1990)	6	0.0015	0.0015	2.500	2.500		0.046		1.46	0.073	0.095	0.001	0.00703	0.117	0.331
Shimizu & Tsujimotot (1994)	13	0.001	0.0015	2.500	10.000	0.041	0.046		1	0.0631	0.1052	0.001	0.00886	0.096	0.331
Dunn *et al.* (1996)	12	0.0064	0.0064	42	384		0.12	1.13	1.13	0.164	0.335	0.00359	0.01607	0.308	0.854
Meijer (1998)	48	0.008	0.008	64	256	0.45	1.5	0.96	1	0.99	2.5	0.00055	0.00205	0.175	1.242
Stone & Shen (2002)	92	0.00318	0.0127	173	696	0.124	0.124	0.96	1.11	0.151	0.314	0.00009	0.04402	0.017	0.502
Poggi *et al.* (2004)	5	0.004	0.004	67	1.072		0.12	1.5	1.5	0.6	0.6	0.00004	0.00032	0.3	0.313
Murphy *et al.* (2007)	24	0.006	0.0064	250	800		0.0139	0.61	1	0.088	0.467	0.00000055	0.000596	0.013	0.197
Fenzl (1962)	26	0.00238	0.00238	11	1.808	0.051	0.152	1.01	1.17	0.058	0.181	0.00163	0.00285	0.052	0.289
Kouwen *et al.* (1969)	27	0.005	0.005	5.000	5.000	0.05	0.1		3	0.149	0.4	0.0005	0.01001	0.03	0.609
Ree & Crow (1977)	30	0.005	0.005	1.464	1.076	0.203	0.305		1	0.242	0.751	0.00042	0.0021	0.046	0.427
Murota *et al.*(1984)	8	0.00024	0.00024	4.000	4.000	0.048	0.06	2.75	2.75	0.092	0.116	0.0005	0.00383	0.085	0.258
Tsujimoto *et al.* (1991)	12	0.0015	0.0015	2.500	2.500	0.0238	0.0419	3.14	3.14	0.07	0.11	0.001	0.007	0.0945	0.4836
Tsujimoto *et al.* (1993)	12	0.00062	0.00062	10.000	10.000	0.061	0.065		2	0.1	0.16	0.00051	0.01076	0.078	0.385
Ikeda & Kanazawa (1996)	7	0.00024	0.00024	20.000	20.000	0.04	0.045		1	0.142	0.19	0.00247	0.00641	0.349	0.606
Meijer (1998)	7	0.0057	0.0057	254	254	1.55	0.165	1.81	1.81	1.75	2.5	0.00109	0.00208	0.142	0.393
Rowinski & Kubrak (2002)	8	0.000825	0.000825	2.500	10.000	0.165	0.165	1.22	1.35	0.1962	0.2475	0.0087	0.0174	0.1587	0.2948
Järvelä (2003)	12	0.0028	0.003	12.000	512	0.155	0.295		1	0.306	0.7065	0.0002	0.0051	0.072	0.33
Carollo *et al.* (2005)	80	0.0045	0.0045	28.000	44.000	0.044	0.082		1	0.061	0.272	0.001	0.05	0.211	1.047

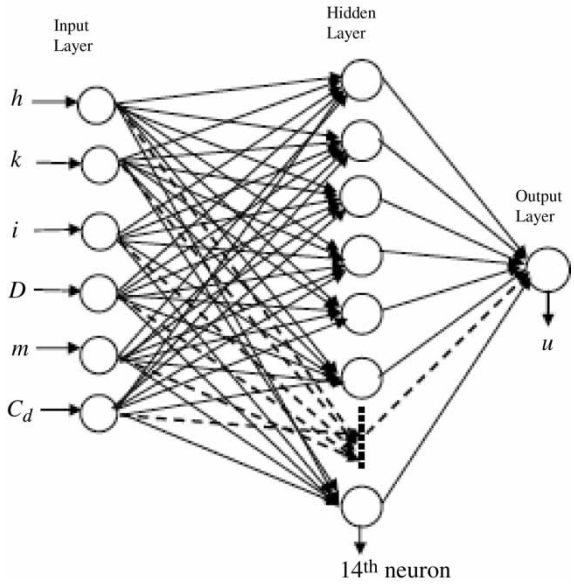

Figure 2 | Optimal artificial neural network.

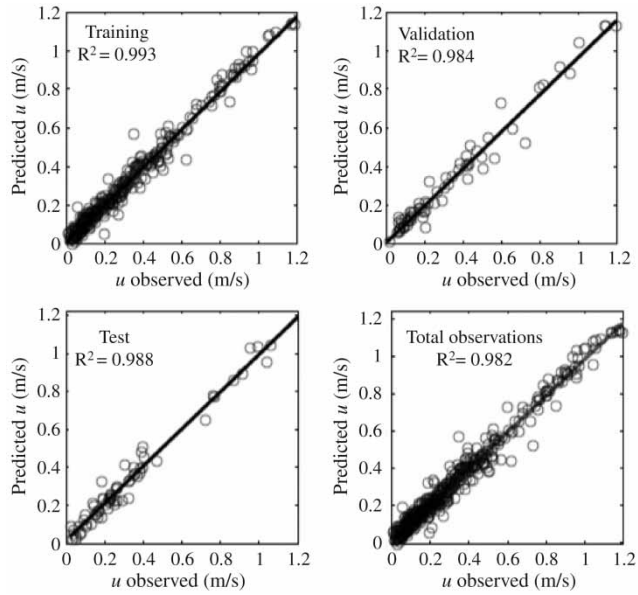

Figure 3 | Results of neural network model.

the products across all hidden neurons. The relative importance of input parameter j is determined through the following formula:

$$\mathrm{Imp}(j) = \sum_{x=1}^{n} CW_{jh(x)} CW_{ho(x)} \tag{8}$$

Table 3 | Comparative analysis of different intelligent predictors

	R^2	
Methodology	**Training**	**Testing**
GA-ANN	0.993	0.988
ANN-LM	0.965	0.92
ANN-BFGS	0.94	0.88
Multiple regression	0.88	0.81

where Imp(j) is the relative importance of parameter j, n the total number of hidden nodes, x the index number of hidden node, $\mathrm{CW}_{jh(x)}$ the connectivity weight between input parameter j and hidden node x, $\mathrm{CW}_{ho(x)}$ is the connectivity weight between hidden node x and the output node. The methodology for Garson's algorithm is as follows:

(a) For each hidden neuron H, divide the absolute value of the input–hidden layer connection weight by the sum of the absolute value of the input–hidden layer connection weight of all input neurons.

For $H = 1$ to nH, for $j = 1$ to nj,

$$A_{ih} = \frac{|W_{jH}|}{\sum_{i=1}^{nj} |W_{jH}|} \tag{9}$$

(b) For each input neuron j, divide the sum of the A j_{H} for each hidden neuron by the sum for each hidden neuron of the sum for each input neuron of A j_{H}, multiply by 100. The relative importance of all output weights attributable to the given input variable is then obtained. j: 1 to nj

$$RI(\%)_j = \frac{\sum_{n=1}^{nh} A_{jH}}{\sum_{n=1}^{nh} \sum_{j=1}^{nj} A_{jH}} \times 100 \tag{10}$$

Connection weight and Garson's algorithm are based on the weight matrices of the neural network. Partial derivative method computes the partial derivatives of the ANN output with respect to the input neurons by making use of the Taylor expansion to approximate the output perturbations with respect to either the input or weight perturbations. Calculations of all approaches for input significance testing are tabulated in Table 4. The values depicted in Table 4 have been derived from both initial and final weight values of the ANN model. As can be seen from Table 4, flow depth

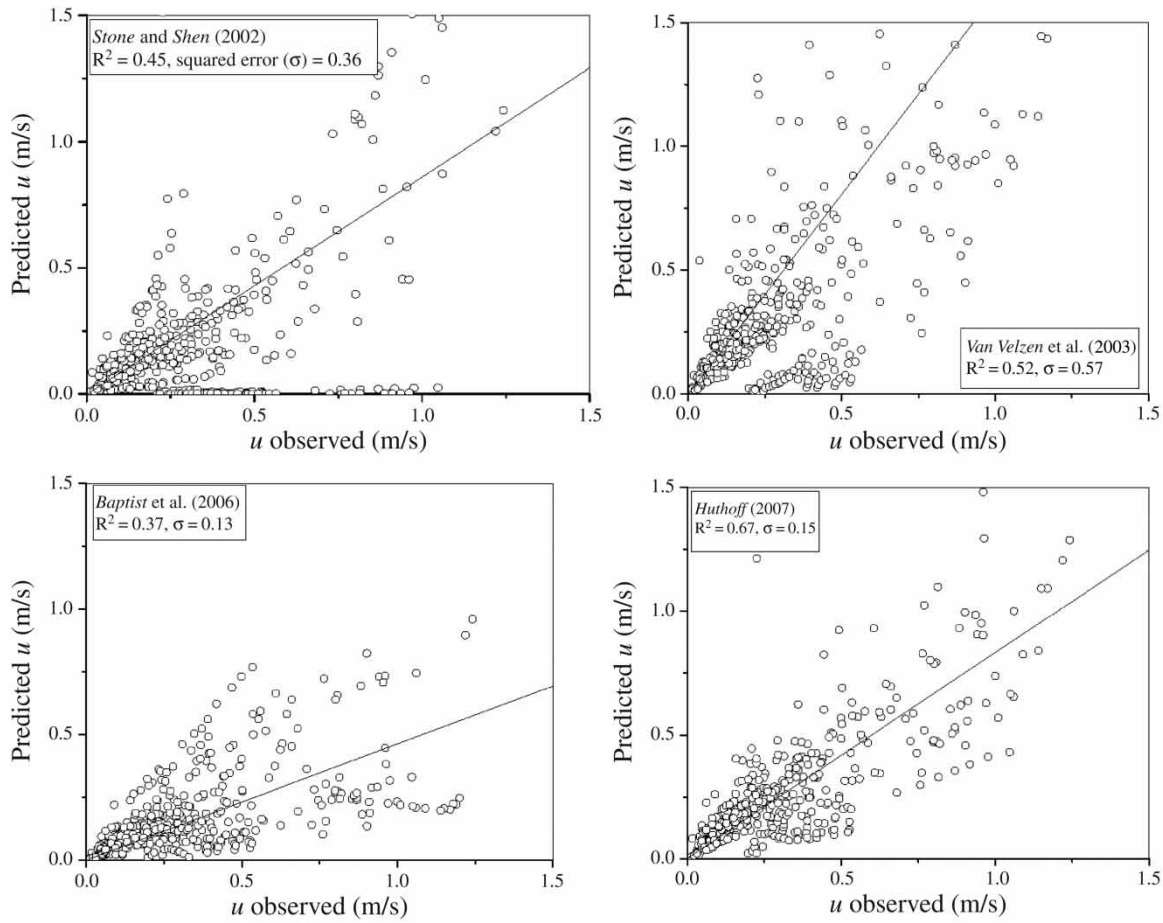

Figure 4 | Performance analysis of flow velocity predictors in vegetative channel.

Table 4 | Input significance and ranking of the variables

	Connection weights		Garson's algorithm		Partial derivatives	
Variables	**Initial weights**	**Final weights**	**Initial weights**	**Final weights**	**Initial weights**	**Final weights**
i	0.08 (6)	0.01 (6)	7.05 (5)	9.12 (6)	0.03 (6)	0.02 (6)
m	0.165 (4)	0.13 (5)	19.1 (3)	15.33 (4)	0.165 (4)	0.12 (5)
k	0.163 (5)	0.15 (4)	4.4 (6)	11.58 (5)	0.09 (5)	0.14 (4)
D	0.164 (3)	0.19 (3)	9.45 (4)	19.86 (3)	0.17 (3)	0.27 (2)
h	0.171 (2)	0.23 (2)	30.5 (1)	23.02 (1)	0.28 (1)	0.15 (3)
C_d	0.25 (1)	0.28 (1)	29.5 (2)	21.03 (2)	0.25 (2)	0.29 (1)

The value in brackets shows rank importance.

and drag coefficient have much more influence on the mean velocity of a vegetative channel. For the channel with densely distributed vegetation, the drag of vegetation becomes the major parameter to characterize the channel hydrodynamics. Turbulent vortices are created in the wakes downstream of the protruding stems in a vegetative channel (Akilli & Rockwell 2002). The associated energy losses of the mean flow field cause the flow to slow down.

These drag effects can become very important, which is also shown in Table 4. Calculations shown in Table 4 indicate that any error in determining flow depth and drag coefficient may result in under- or over-estimation of mean velocity. Table 4 also highlights the importance of D on mean velocity prediction.

Contribution plots for each of the predictor variables have been constructed in the present work, as shown in Figure 5. Figure 5(a) shows the variation of u with h and i by keeping the other variables at mean value. The range of i has been divided into 10 parts, and plots of h and u have been drawn at each value of i. It can be seen from Figure 5(a): (i) mean velocity increases with an increase in h and (ii) it also increases with an increment in the channel slope (i). Mean velocity is directly proportional to flow depth and channel slope as per any flow equations existing in the literature. It is probably more relevant to analyze the trends shown in Figure 5(a) through Manning's formulation as it is most often employed in practice. The trend shown in Figure 5(a) matches with the most used Manning's formula for mean velocity. Mean velocity will increase with flow depth and channel slope as per Manning's formula. Manning's formula, however, has Manning's coefficient as controlling factor, thereby limiting the increasing trend of mean velocity with flow depth and channel slope. The same can be seen in Figure 5(a). When flow depth and channel slope have high values, increment in mean velocity is slow and it tends towards attaining an asymptotic nature. Figure 5(b) shows the combined variation of D and k with u. D and k are descriptors of vegetation properties. Except for Stone and Shen's equation (Stone & Shen 2002), most of the equations given in Table 1 show that u will decrease non-linearly with an increase in D. Equations given in Table 1 do not show a general behavior of k with u; however, Figure 5(b) shows that u will increase with k. The trend that u will decrease non-linearly with an increase in D is also shown in Figure 5(c). m represents the density of vegetation. When m is very high, flow will be very low because of high density of vegetation, which is shown in Figure 5(c). Drag opposes the fluid motion; this general behavior is shown in Figure 5(d). u will decrease with an increase in C_d, which may be due to increasing moment absorbing area. The non-linearity, shown in Figures 5(a) to 5(d), probably originates from attenuation effects of vegetation that reduce bed surface contribution to the overall resistance (Armanini *et al.* 2005).

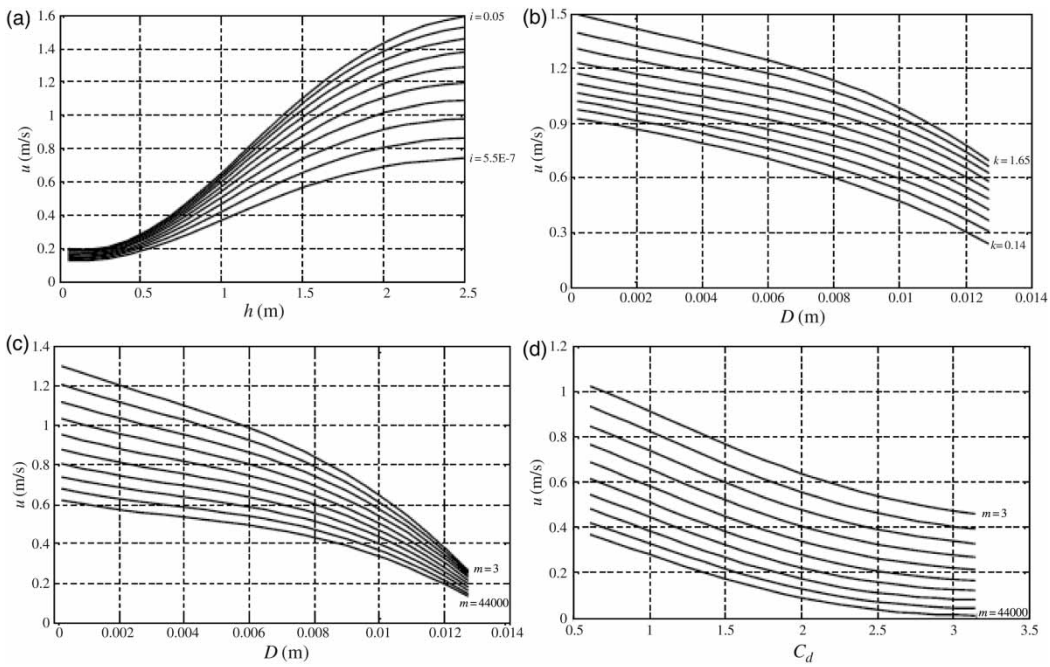

Figure 5 | Variability of u with input parameters: (a) relationship between u, h, and i; (b) variation of D and k with u; (c) variation of D and k with u; (d) relationship between u, C_d, and m.

CONCLUSION

Vegetation–flow interactions are central to many problems of practical interest to hydraulic engineers, including flood risk studies, sediment transport studies, and the analysis of the hydraulic performance of river restoration schemes. Existing predictors are too complex and limited in terms of their applicability. Based on a large database of submerged vegetation conditions (rigid and flexible), the present work formulates a neural network model of mean velocity in a vegetative channel comprising all the relevant parameters. In this way, a qualitative effect of all types of parameters can be incorporated in the prediction of flow velocity in a vegetative channel. The generalization capacity of the model is very good considering there are so many different parameters affecting flow–vegetation interactions. In engineering applications, errors in determining the governing parameters affect the accuracy of the prediction. The present analysis based on the input significant test shows that in order to get the accurate flow velocity prediction, there should be more accurate determination of flow depth and drag coefficient due to vegetation. The variability of flow velocity with different parameters is also important to understanding the process. The present work shows the mean velocity decreases with an increase in vegetation diameter, vegetation density, and drag coefficient. The variability of mean velocity with vegetation height is an increasing one.

ACKNOWLEDGEMENT

The author gratefully acknowledges the financial support received from the Department of Science and Technology, Government of India (SERC-DST: SR/S3/MERC/005/2010) to carry out the research work presented in this paper.

REFERENCES

Akilli, H. & Rockwell, D. 2002 Vortex formation from a cylinder in shallow water. *Phys. Fluids* **14**, 2957–2967.

Armanini, A., Righetti, M. & Grisenti, P. 2005 Direct measurement of vegetation resistance in prototype scale. *J. Hydraul. Res.* **43**, 481–487.

Baptist, M. J., Babovic, V., Uthurburu, J. R., Keijzer, M., Uittenbogaard, R. E., Verway, A. & Mynett, A. E. 2006 On inducing equations for vegetation resistance. *J. Hydraul. Res.* **45**, 435–450.

Bhattacharya, B., Price, R. K. & Solomatine, D. P. 2004 A data mining approach to modelling sediment transport. In: *6th International Conference on Hydroinformatics* (S.-Y. Liong, K.-K. Phoon & V. Babovic, eds), 21–24 June, World Scientific Publishing Company, Singapore.

Borovkov, V. S. & Yurchuk, M. 1994 Hydraulic resistance of vegetated channels. *Hydrotech. Const.* **28**, 432–438.

Carollo, F. G., Ferro, V. & Termini, D. 2005 Flow resistance law in channels with flexible submerged vegetation. *J. Hydraul. Res.* **131**, 554–564.

Chen, S., Wu, Y. & Luk, B. L. 1999 Combined genetic algorithm optimization and regularized orthogonal least squares learning for radial basis function networks. *IEEE Trans. Neural Netw.* **10**, 1239–1243.

Cheng, C. T., Ou, C. P. & Chau, K. W. 2002 Combining a fuzzy optimal model with a genetic algorithm to solve multi-objective rainfall-runoff model calibration. *J. Hydrol.* **268** (3), 72–86.

Choi, S. U. & Kang, H. 2004 Reynolds stress modeling of vegetated open-channel flows. *J. Hydraul. Res.* **42**, 3–11.

Dolling, O. R. & Varas, E. A. 2002 Artificial neural networks for streamflow prediction. *J. Hydraul. Res.* **40**, 547–554.

Duh, M. S., Walker, A. M. & Ayanian, J. Z. 1998 Epidemiologic interpretation of artificial neural networks. *Am. J. Epidemiol.* **147**, 1112–1122.

Dunn, C., Lopez, F. & García, M. 1996 Mean flow and turbulence in a laboratory channel with simulated vegetation. Hydraulic Engineering Series. 51. University of Illinois at Urbana-Champaign, Urbana, IL.

Einstein, H. A. & Banks, R. B. 1950 Fluid resistance of composite roughness. *American Geophysical Union* **3**, 603–634.

Fenzl, R. N. 1962 Hydraulic resistance of broad shallow vegetated channels. PhD thesis, University of California, Davis, CA.

Galema, A. 2009 Evaluation of Vegetation Resistance Descriptors for Flood Management. Master Thesis, University of Twente, Enschede, The Netherlands.

Garson, G. D. 1991 Interpreting neural-network connection weights. *Artif. Intell. Expert.* **6**, 47–51.

Giustolisi, O. & Savic, D. A. 2006 A symbolic data-driven technique based on evolutionary polynomial regression. *J. Hydroinform.* **8** (3), 207–222.

Giustolisi, O. & Simeone, V. 2006 Optimal design of artificial neural networks by multi-objective strategy: groundwater level predictions. *Hydrol. Sci. J.* **51**, 502–523.

Goldberg, D. E. 1989 *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, Reading, MA.

Green, J. C. 2006 Effect of macrophyte spatial variability on channel resistance. *Adv. Water Res.* **29**, 426–438.

Haykin, S. 1999 *Neural Networks: A Comprehensive Foundation*. Prentice-Hall, Upper Saddle River, NJ.

Huthoff, F. 2007 Modelling Hydraulic Resistance of Floodplain Vegetation. PhD Thesis, Department of Water Engineering, University Twente, Enschede, The Netherlands.

Huthoff, F., Augustijn, D. C. M. & Hulscher, S. J. M. H. 2007 Analytical solution of the depth-averaged flow velocity in

case of submerged rigid cylindrical vegetation. *Water Resour. Res.* **43**, W06413.

Ikeda, S. & Kanazawa, M. 1996 Three-dimensional organized vortices above flexible water plants. *J. Hydraul. Eng.* **122**, 634–640.

Irie, B. & Miyanki, S. 1988 Capabilities of three layer perceptrons. In *IEEE Second International Conference on Neural Networks*, San Diego, CA, 1, pp. 641–648.

Järvelä, J. 2002 Flow resistance of flexible and stiff vegetation: a flume study with natural plants. *J. Hydrol.* **269**, 44–54.

Järvelä, J. 2003 Flow resistance of flexible and stiff vegetation: a flume study with natural plants. *J. Hydrol.* **269**, 44–54.

Koza, J. R. 1992 *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, Cambridge, MA.

Kouwen, N. & Fathi-Moghadam, M. 2000 Friction factors for coniferous trees along rivers. *J. Hydraul. Eng.* **126**, 732–740.

Kouwen, N., Unny, T. E. & Hill, H. M. 1969 Flow retardance in vegetated channels. *J. Irrig. Drain. E.* **95** (2), 329–344.

Kumar, B. 2011 Data mining approach for friction factor in mobile bed channel. *Water Manage.* **164**, 15–25.

Lopez, F. & Garcia, M. 1998 Open-channel flow through simulated vegetation: Suspended sediment transport modeling. *Water Resour. Res.* **34**, 2341–2352.

Masters, T. 1993 *Practical Neural Network Recipes in C++*. Academic Press, San Diego, CA.

Meijer, D. G. 1998 Modelproeven overstroomd riet. Technical report PR177, HKV Consultants, Lelystad, The Netherlands.

Millar, R. G. 2002 Influence of bank vegetation on channel properties. *Water Resour. Res.* **36**, 1109–1118.

Montgometry, D. R. & Piegay, H. 2003 Woods in rivers: interactions with channel morphology and processes. *Geomorphology* **51**, 1–5.

Murota, A., Fakuhara, T. & Sato, M. 1984 Turbulence structure in vegetated open channel flow. *J. Hydrosci. Hydraul. Eng.* **2**, 47–61.

Murphy, E., Ghisalberti, M. & Nepf, H. 2007 Model and laboratory study of dispersion in flows with submerged vegetation. *Water Resour. Res.* **43**, W05438.

Nasseri, M., Asghari, K. & Abedini, M. J. 2008 Optimized scenario for rainfall forecasting using genetic algorithm coupled with artificial neural network. *Expert Syst. Appl.* **35**, 1415–1421.

Neary, V. S. 2003 Numerical solution of fully-developed flow with vegetative resistance. *J. Eng. Mech.* **129**, 558–563.

Nepf, H. M. 1999 Drag, turbulence, and diffusion in flow through emergent vegetation. *Water Resour. Res.* **35**, 479–489.

Ng, W. Y., Yeung, D. S., Wang, X. & Cloete, I. 2004 A study of the difference between partial derivative and stochastic neural network sensitivity analysis for applications in supervised pattern classification problems. *Proceedings of the Third International Conference on Machine Leaming and Cybernetics*, Shanghai, 4283–4288.

Nikora, V., Larned, S., Nikora, N., Debnath, K., Cooper, G. & Reid, M. 2008 Hydraulic resistance due to aquatic vegetation in small streams: a field study. *J. Hydraul. Eng.* **134**, 1326–1332.

Olden, J. D. & Jackson, D. A. 2002 Illuminating the 'blackbox': understanding variable contributions in artificial neural networks. *Ecol. Model.* **154**, 135–150.

Poggi, D., Porporato, A., Ridolfi, L., Albertson, J. D. & Katul, G. G. 2004 The effect of vegetation density on canopy sub-layer turbulence. *Boundary-Layer Meteorol.* **111**, 565–587.

Ree, W. O. & Crow, F. R. 1977 Friction factors for vegetated waterways of small slope. Technical Report Publication S-151. US Department of Agriculture, Agricultural Research Service.

Rowinski, P. M. & Kubrak, J. 2002 A mixing-length model for predicting vertical velocity distribution in flows through emergent vegetation. *Hydrol. Sci. J.* **47**, 893–904.

Salajegheh, E. & Gholizadeh, S. 2005 Optimum design of structure by an improved genetic algorithm using neural networks. *Adv. Eng. Softw.* **36**, 757–767.

Shields, F. D. & Cooper, C. M. 2000 Woody vegetation and debris for in-channel sediment control. *Int. J. Sed. Res.* **15**, 83–92.

Shimizu, Y. & Tsujimoto, T. 1994 Numerical analysis of turbulent open-channel flow over a vegetation layer using a k-e turbulence model. *J. Hydrosci. Hydraul. Eng.* **2**, 55–67.

Singh, A. K., Deo, M. C. & Kumar, V. S. 2007 Neural network–genetic programming for sediment transport. *Maritime Eng.* **160**, 113–119.

Srinivasulu, S. & Jain, A. 2006 A comparative analysis of training methods for artificial neural network rainfall–runoff models. *Appl. Soft Comput.* **6**, 295–306.

Stephan, U. & Gutknecht, D. 2002 Hydraulic resistance of submerged flexible vegetation. *J. Hydrol.* **269**, 27–43.

Stone, B. M. & Shen, H. T. 2002 Hydraulic resistance of flow in channels with cylindrical roughness. *J. Hydraul. Eng.* **128**, 500–506.

Tsujimoto, T. & Kitamura, T. 1990 Velocity profile of flow in vegetated bed channels. KHL progressive report 1, Kanazawa University, Kanazawa, Japan.

Tsujimoto, T., Okada, T. & Kitamura, T. 1991 Turbulent flow over flexible vegetation covered bed in open channels. KHL progressive report 1, Kanazawa University, Kanazawa, Japan.

Tsujimoto, T., Okada, T. & Kontani, K. 1993 Turbulent structure of open channel flow over flexible vegetation. KHL progressive report 4, Kanazawa University, Kanazawa, Japan.

Van Velzen, E. H., Jesse, P., Cornelissen, P. & Coops, H. 2003 Stromingsweerstand vegetatie in uiterwaarden. Handbook report 2003.028, RIZA, Arnhem, The Netherlands.

Wang, X., Lu, W., Cao, S. & Fang, D. 2007 Using time-delay neural network combined with genetic algorithms to predict runoff level of Linshan watershed, Sichuan, China. *J. Hydrol. Eng.* **12** (2), 231–236.

Wei, C. 2013 Soft computing techniques in ensemble precipitation nowcast. *Appl. Soft Comput.* **13**, 793–805.

Wilson, C. A. M. E., Stoesser, T., Bates, P. D. & Batemann-Pinzen, A. 2003 Open channel flow through different forms of submerged flexible vegetation. *J. Hydraul. Eng.* **129**, 847–853.

Yen, B. C. 2002 Open channel flow resistance. *J. Hydraul. Eng.* **128**, 20–39.

First received 6 May 2013; accepted in revised form 28 September 2013. Available online 22 November 2013