World Applied Sciences Journal 5 (6): 646-654, 2008
ISSN 1818-4952

# Application of Artificial Neural Network in Estimating the Fall Velocity of Sediment Particles

[1]*Seyed Morteza Sadat-Helbar,* [1]*Ebrahim Amiri-Tokaldany,* [2]*Fatemeh Darzi and* [3]*Steve Darby*

[1]Department of Irrigation and Reclamation Engineering,
Faculty of Water and Soil Engineering, University of Tehran, Iran
[2]Department of Water Engineering, Faculty of Civil Engineering, University of Esfehan, Esfehan, Iran
[3]School of Geography, University of Southampton, Southampton So17 1BJ, UK

**Abstract:** The fall velocity of sediment particles is one of the important parameters in area dealing with sediment transport. Many attempts to estimate the fall velocity have been carried out by researchers, mostly during last 40 years, so there are a large number of relations introduced to apply for different particle sizes in various conditions. Availability of these considerable relations in number, however, confuses the engineers to make a right decision on using the suitable relation to estimate fall velocity. In this research, using Artificial Neural Network (ANN) a method is developed to estimate the fall velocity of natural sediment particles. The ANN used in this research, is designed and validated using 115 series of measured data, reported by different researchers. The multi layer perceptron network with quick back propagation learning scheme was used to recover the nonlinear mapping between input data (independent variables) and output of the network (dependent variable). This nonlinear mapping is used to intelligent estimation of fall velocity. To evaluate the predicting precision of the model, the prediction of the designed network were compared with results of 14 experimental and analytical models of previous researches. It is found that ANN predicts better results than available models.

**Key words:** Fall velocity · Sediment particles · Reynolds number · Relative density · Artificial neural network

## INTRODUCTION

The qualitative analysis of sediment transports in river engineering problems, such as sedimentation in river courses and morphological changes of river banks, designing the settling basins of water conveyance networks and sedimentation of dam reservoirs, needs to use a suitable relation to estimate the terminal fall velocity, sometimes called settling velocity, of sediment particles. The terminal fall velocity of a particle is the particle downward velocity in a low dense fluid at equilibrium in which the sum of the gravity force, buoyancy force and fluid drag force being equal to zero. Fall velocity of a particle, depends on the density and viscosity of the fluid and the density, size, shape, spherically and the surface texture of the particle. Many attempts to predict the particle fall velocity have been carried out by researches, started by Stokes in 1851 (cited in [6]) and followed by Oseen [1], Rubby [2], Rouse [3], Interagency Committee [4], Zanke [5], Yallin [6], Hallermier [7], Dietrich [8], Van Rijn [9], Concharov [cited in [12]], Julien [10], Cheng [11], Jimenez and Madsen [12], Brown and Lawler [13], She *et al.* [14] and Wu and Wang [15] among others, who developed empirical or semi-empirical relations for estimating the settling velocity of sediment particles.

Most of above mentioned investigations, however, have some limitations when it comes to applying them to engineering works. For instance, the relations developed by Stokes (cited in [6]), Rouse [3], Brown and Lawler [13], are applicable only to spherical particles. Even for spherical particles, the analytical solution of Stokes is only applicable for Reynolds number less than 1 and there is no analytical solution to predict the fall velocity of natural particles. In the absence of such a solution, some laboratory investigations have been conducted to provide design curves to predict the fall velocity based solely on the diameter of standard particles [2,14,29]. A family curves, also, were provided to predict the effects of other particle characteristics on fall velocity; e.g. Alger and Simons [16], Komar and Reimers [17] among others.

In a useful attempt, the US Interagency Committee on Water Resources summarized the data obtained by several researchers and published a graphical relation to estimate drag coefficient and consequently, to calculate

**Corresponding Author:** Seyed Morteza Sadat-Helbar, Department of Irrigation and Reclamation Engineering,
Faculty of Water and Soil Engineering, University of Tehran, Iran

settling velocity (cited in [26]). This graph, however, includes a series of curves and tables and several interpolations must be conducted to obtain the answer, making it inconvenient to use. Recently, Wu and Wang [15] have re-evaluated the relation of US Interagency Committee using a wide range of data and using the equation proposed by Cheng (cited in [26]), introduced an explicit mathematical expression for the settling velocity of natural sediment particles. Wu and Wang [15] reported that by considering the effects of viscosity and Corey shape factor, their formula has a relative mean error of 9.1% which decrease to a relative mean error of 6.8% when the effects of Corey shape factor are neglected. They concluded that their relation performed better than nine existing formula in the literature.

In this research, by using a computational approach inspired by the human nervous system, i.e. Artificial Neural Network, a new method to estimate the fall velocity is presented.

**Development of New Model:** In 1851 Stokes by using Navier-Stokes equations, along with continuity equation expressed in polar coordinates, investigated the coefficient of drag applies by fluid flow upon a spherical particle [30]. Based on Stokes' results, the fall velocity of spherical particles in region of particle Reynolds number ($R_e$) less than 1, can be calculated using [11]:

$$w = \frac{1}{18}\frac{g(s-1)d^2}{\nu} \qquad (1)$$

In which $w$ = particle fall velocity in m/s, $g$ = acceleration due to gravity in m/s$^2$, $d$ = particle diameter in m, $\nu$ = kinematic viscosity in m$^2$/s and $s$ = relative density ($\rho_s/\rho$) where $\rho_s$ and $\rho$ = the density of sediment particle and fluid in t/m$^3$, respectively.

For natural sediment particles, many researches have attempted to develop similar equation. Due to extensive variation of natural particles' geometry, however, there has been a little success in this regard, so that a large number of different equations, each of which can only be applied to a limit range of sediment and fluid conditions, have been developed. In this research, famous relations of fall velocity introduced from 1933 to 2006, have been collected and their advantages and limitations have been investigated. Based on this, for the purpose of comparison, 14 relations have been chosen. The method of developing the new relation is presented in section followed by a description of the basis of the Artificial Neural Network.

## Artificial Neural Networks (ANN)

**The Basics:** An ANN is a computational approach inspired by the human nervous system. Its data processing paradigm is made up of highly interconnected nodes (neurons) that map a complex input pattern with a corresponding output pattern [18,19]. The artificial neural networks are massively parallel distributed processing and computing techniques inspired by biological neuron processing. The universal approximator, artificial neural networks mimics the function of human baring by acquiring knowledge through process of learning. The ability to gather knowledge through the process of learning, like a human brain, from sufficient predictor patterns makes it possible to apply the ANN to solve large-scale real world problems.

Once the ANN is trained, the relationship between the predictor (input) and predicted (output) variables is encoded in the network. Then it can be used to predict the output based on the information fed to the input nodes. ANN is used to define the network topology as well as to simulate the learning, validation and testing phases without imposing any functional relationships between the independent and dependent variables. With this architecture, ANN methodology has proven to be a powerful mathematical model, which excels at function approximation and pattern recognition. Added to that, it is more robust and flexible than other type of mathematical models (Cited in [18]). It is worth to mention that the predictive efficiency of an ANN model is largely dependent on the architecture of ANN model.

The most popular neural network model is the Multi-layered Perceptron (MLP) [20]. In this type of network, the artificial neurons, or processing units, are arranged in a layered configuration containing an input layer, usually one hidden layer and an output layer (Fig. 1). The MLP is a layered feed-forward network, which is typically trained with static Back Propagation (BP). It is simple, robust and very powerful in pattern recognition, classification and mapping. MLP is capable of approximating any measurable function from one finite-dimensional space to another within a desired degree of accuracy [21].

Units in the input layer introduce normalized or filtered values of each input into the network. Units in the hidden and output layers are connected to all of the units in the preceding layer. Each connection carries a weighting factor $W$ and each neuron has a scalar bias, $b$. In a feed-forward network, the input quantities are fed to input nodes, which in turn pass them on to the hidden layer nodes after multiplying by a given weight. A hidden layer node adds up the weighted input received from each

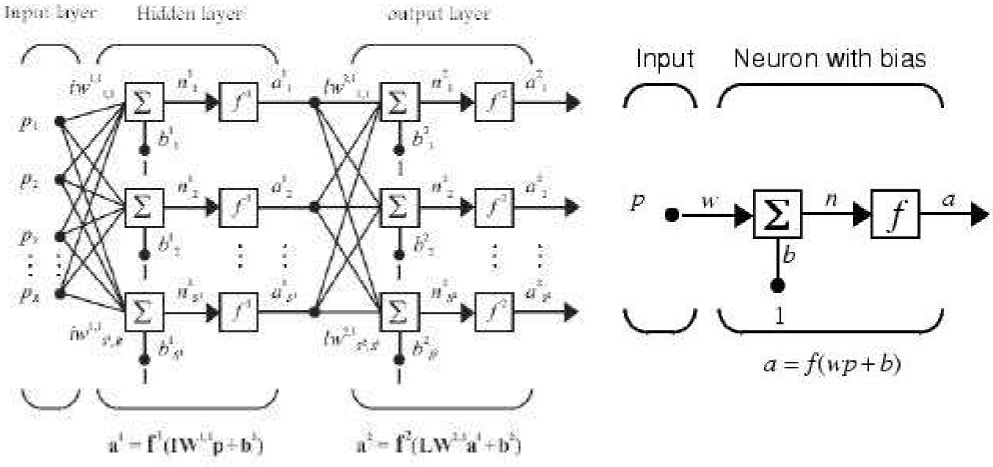
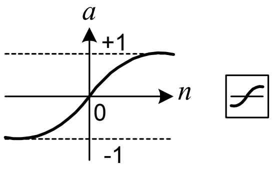


Fig. 1: Multi-Layered Perceptron (MLP)



$a = tan\,sig\,(n)$

Fig. 2: Tan-sigmoid transfer function

input node, associates it with a bias and then passes the result through a mathematical transfer function *f*, to create an output signal that is sent to processing units in the next layer. Neurons may use any differentiable transfer function to generate their output. The most commonly used transfer functions in multi-layered networks are tan-sigmoid (Fig.2), log-sigmoid and linear transfer function. Since the sigmoid transfer function is used in the model, the input-output data have been scaled appropriately to fall within the function limits.

**Model Structure Identification:** There are multitudes of network types available for ANN applications and its choice depends on the nature of the problem and data availability. The Multi-Layered Perceptron (MLP) trained with the back propagation algorithm is perhaps the most popular network for hydraulic and hydrologic modeling.

Before its application to any problem, the network is first trained, whereby the target output at each output node is compared with the network output and the difference or error is minimised by adjusting the weights and biases through some training algorithm.

This algorithm first computes the error signal at the output layer and then it is propagated to the input layer through hidden layer(s). After computing error signals, the algorithm adjusts the synaptic weights between input and hidden layers. These procedures will continue till the error between target output and the model output is less than the specified permissible value.

Standard back propagation is a gradient descent algorithm, in which the network weights are moved along the negative of the gradient of the performance function. The term “back propagation” refers to the manner in which the gradient is computed for nonlinear multi-layered networks. The procedure usually involves minimization of some cost function such as mean of square errors given by equation (2) or sum of square errors.

$$E = \left( \sum_{P} \sum_{p} (y_i - t_i)^2 \right) \times \frac{1}{P} \qquad (2)$$

where , $y_i$ = ANN output, $t_i$ = desired output, $p$ = number of output nodes and $P$ = number of training patterns or data sets.

Training an ANN is a mathematical exercise that optimizes all of the ANN’s weights and threshold values, using some fraction of the available data. One of the problems that occurs during neural network training is over-fitting. The error on the training set is driven to a very small value. When, a new data is presented to the network, however, the error is large. The network has memorized the training examples, but it has not learned to generalize to new situations. A most common method for improving generalization is called early stopping. In this technique the available data is divided into three sub-sets. The first subset is the training set, which is used for computing the gradient and updating the network weights and biases. The second subset is the validation set. The error on the validation set is monitored during the training process. The validation error will normally decrease during the initial phase of training, as does the training set error. However, when the network begins to overfit the data, the error on the validation set will typically begin to rise. When the validation error increases for a specified number of iterations, the training is stopped and the weights and biases at the minimum of the validation error are returned. The test set error, as the third sub-set, is not used during the training, but it is used to compare different models.

The important step in ANN modelling is to find the optimal number of neurons in the hidden layer. Optimization routines can be used to determine the ideal number of units in the hidden layer and the nature of their transfer functions. These routines often involve considerable trial and error procedures, increasing the number of nodes and monitoring the performance of the ANN. If the ANN starts exhibiting noisy fluctuations, it is being over-trained or exhibits a high variance in the network. The number of nodes in the hidden layer would satisfy the following condition:

$$(A+1)B+(B+1)C \leq \frac{1}{10} \times D \tag{3}$$

where:
A = numbers of nodes in input layer,
B = numbers of nodes in hidden layer,
C = numbers of nodes in output layer,
D = numbers of training basis.

In most cases, increasing the number of nodes will improve the performance of MLP on the training data, but not necessarily on the validation data. The number of neurons in input and output layers is fixed, so only the neurons in the hidden layer are determined. If enough hidden units are added, hence it will have enough weights to exactly represent all the training patterns which in turn the network will get its training set 100% correct. This would, however, be extremely poor networks because it would have little ability to generalize or find solutions for examples that it had not been trained on.

The correct way of assessing the impact of the number of hidden units on a problem, is to look at the performance on the validation set. As the total number of hidden units are increased from one, the network performance on the validation data increases rapidly. This is because each new hidden unit starts to represent one of the underlying features in the data set. As more units are added, performance levels off. Adding further units may then cause a decrease in performance because the power of generalization is lost and the network begins to learn the noise present in the data. By using a measure of the error in the validation set as part of its convergence criteria, Neural Connection reduces the danger of over learning. Nonetheless, it is still best to use as few nodes as possible to achieve the desired result (cited in [18]).

**ANN Model Development:** The goal of an ANN model is to generalize a relationship of the form of:

$$Y_m = f(X_n) \tag{4}$$

where:
$X_n$= an n-dimensional input vector consisting of variables $x_1,..., x_i,...,$ $x_n$; $Y_m$ = an m-dimensional output vector consisting of the resulting variables of interest $y_1,..., y_i,...,$ $y_m$. In sediments fall velocity modeling, values of $x_i$ may include nominal diameter in m ($d_N$), kinematic viscosity in m$^2$/s ($\upsilon_s$), relative density ($s$) and Corey shape factor ($S_f$) and the value of $y_i$ represent the particle fall velocity in m/s ($w$). Therefore, the network has four neurons in the first layer and one neuron in the third output layer. Relationship between input and output may be expressed as:

$$w = f(s, S_f, \nu_s, d_N) \tag{5}$$

In this research, a standard back propagation algorithm, i.e. Levenberg-Marquardt algorithm, is employed for training a single hidden layer feed-forward ANN model. The Levenberg-Marquardt algorithm appears to be the fastest method for training moderate-sized feed-forward neural networks (up to several hundred weights). The model used here, has three neuron layers. The number of neurons in the input and output layers are equal to the number of input and output parameters. The number of neurons in the hidden layer is dependent on the complexity and nonlinearity of the problems. On the basis of trial and error evaluation of the ANN architectures, the number of neurons in the hidden layer is taken as number of neurons in the input layer.

A tangent sigmodial function is used as the transfer function in hidden layer and a purline function in the output layer. A MATLAB program is developed to implement the back propagation algorithms. In Table 1, a brief summary of ANN Model Structure is shown.

Table 1: ANN Model Structure

| Transfer function | | neuron | | |
|---|---|---|---|---|
| Output layer | Hidden layer | Output layer | Hidden layer | Number of layer |
| purelin | tansig | 1 | 4 | 2 |

Table 2: Data used for testing, validation and training ANN model [14].

| | | | d (mm) | | w (m/s) | |
|---|---|---|---|---|---|---|
| Code | Data | n | Range | Mean | Range | Mean |
| 1 | Cheng (1997)[a,c] | 37 | 0.061-4.5 | 1.15 | 0.002-0.281 | 0.101 |
| 2 | Engelund and Hansen (1972)[c,d] | 22 | 0.100-1.9 | 0.580 | 0.005-0.170 | 0.063 |
| 3 | Hallermeier (1981)[b,c,d] | 20 | 0.152-0.61 | 0.369 | 0.017-0.075 | 0.045 |
| 4 | Raudkivi (1990)[d,e] | 12 | 0.200-2.0 | 0.930 | 0.017-0.156 | 0.082 |
| 5 | Raudkivi (1990)[d,f] | 12 | 0.200-2.0 | 0.930 | 0.018-0.194 | 0.098 |
| 6 | Raudkivi (1990)[d,g] | 12 | 0.200-2.0 | 0.930 | 0.019-0.240 | 0.116 |

[a] Arithmetic average diameter ($d_n$).

[b] Assuming $d_s/d_N \approx 0.9$.

[c] Shape factor not given.

[d] Nominal diameter ($d_N$).

[e] Shape factor=0.5.

[f] Shape factor=0.7.

[g] Shape factor=0.9.

To use two polar sigmoidal fuction in hidden layers, the input data should be normalized prior to ANN training, so that the data transformed to values between 0 and 1. The output of this function is numbers between 1 and -1. The form of the input data plays an important role in network learning processes. For data input nearly equal to 0 or 1, due to the form of the two polar sigmoidal function, the elements of performer acts slowly and consequently, the change of neuron weights is minimal in this range of data. For data inputs close to 0.5, the response of the neurons would be faster. Hence, the data normalization was done so that the mean value of data set becomes equal to 0.5, using following equation:

$$X_{norm} = 0.5\left[\frac{X_0 - \overline{X}}{X_{max} - X_{min}}\right] + 0.5 \tag{6}$$

in which $X_0$ = original data, $\overline{X}$ = mean data, $X_{max}$ = maximum data, $X_{min}$ = minimum data and $X_{norm}$ = normalized data.

**Data Sets:** Six data sets, introduced by Jimenez and Madsen [12], are used for testing, validation and training ANN model (Table 2). In Table 2, the number of data points from each source, n, is listed in the third column. The data sets, first, grouped into three groups. The first group, were taken from Cheng [11], Engelund and Hansen [22] and Hallermeir [7], corresponds to the settling velocities of natural sediments without an explicit definition of the shape factor, but taking it as equal to 0.7, as it is usually taken as the most common value for naturally shaped sediments (see [8]).

The Cheng [11] data set is a compilation of Russian quartz sand experiments (original references can be found in [11]) in which the sediment size was characterized through the arithmetic average diameter [11]. Because the specific gravity, *s*, was not given, it was assumed to be 2.65. The kinematic viscosity of the fluid was calculated as corresponding to fresh water at the specified temperature.

The Engelund and Hansen [22] data set was taken from Fredsoe and Deigaard [23]. The sediment size was characterized through the sieve diameter $d_s$ (not used here) and the nominal diameter $d_N$ and settling velocities were measured at 10°C and 20°C. Similar to Cheng's data set, the specific gravity *s* was not given, so it was assumed to be 2.65 and the kinematic viscosity of the fluid was calculated corresponding to fresh water at the specified temperature.

The Hallermeir [7] data set is a compilation of previously published experiments (original refrences can be found in [7]), in which the sediment size was characterized by the sieve diameter.

Since the method proposed here was derived to be used with the nominal diameter, the given sieve diameters were, as previously mentioned, converted to nominal diameter by using the rule of thumb $d_s/d_N \approx 0.9$ [24]. As an example of the applicability of this approach, the ratio calculated from the data supplied by Engelund and Hansen [22] gives a mean value of 0.93 with a standard deviation of 0.04. The analysis in this research is restricted to sands having size in the quartz range. Hence, only experiments with a specific gravity between 2.57 and 2.67 were considered. The kinematic viscosity was given for some experiments and when it was not, a value of $10^{-6}$ m$^2$/s was assumed [7]. The original compilation of Hallermeier's data set also included the Engelund and Hansen [22] data, but it is considered in this research, separately.

The second group of data sets corresponds to the sediment settling velocities reported by Raudkivi [24], originally given by the U.S. Inter-Agency Committee [4]. The reported data consisted of settling velocities of sediment characterized by its nominal diameter and shape factor (Table 2).

Finally, the total set of generated patterns has been divided into three sub-sets immethodically. About 23 patterns are kept aside for validation, 23 patterns for testing and remaining 69 were used for training the neural network.

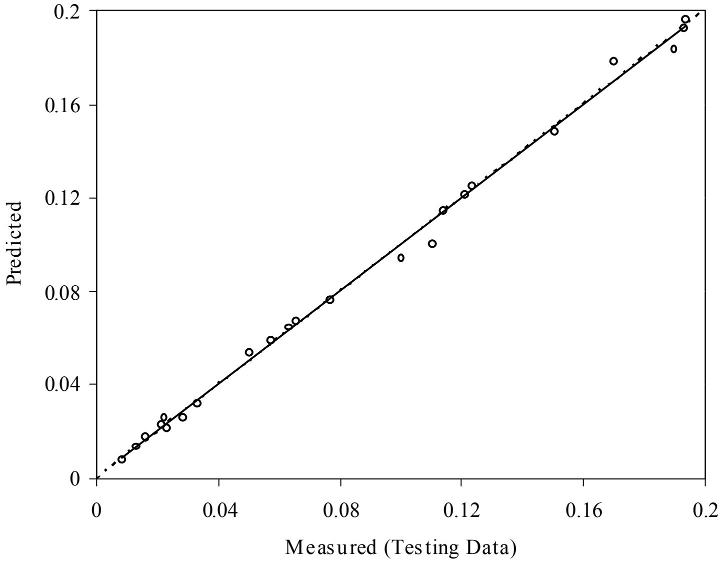


Fig. 3: Measured and predicted fall velocities by ANN according to testing data set

**Performance Evaluation of the ANN Model:** A variety of verification criteria could be used for the evaluation and inter-comparison of different models. They are grouped into two groups, graphical and numerical performance indicators. Coefficient of correlation (R) from first group shows the strength of the relationship between measured and predicted of sediments fall velocity, i.e. high value of the coefficient of correlation (e.g. R>95%) represents a strong relationship between measured and predicted values. The scatter plot of measured and predicted *w* at testing set is shown in Fig. 3. This plot shows the degree of correlation between measured and predicted fall velocities. It can be observed that the correlation between the measured and predicted *w* is high and the scatter plot resemble a straight line with a slope of 1:1.

To describe the accuracy of ANN model quantitatively, the relative error of the relation for each particle size and the mean relative error, are calculated using the following equations:

$$RE_{(di)} = \frac{\left| w_{o(di)} - w_{(di)} \right|}{w_{o(di)}}$$

$$MRE = \frac{1}{n} \sum_{i=1}^{n} RE_{(di)} \qquad (7)$$

where:

$RE_{(di)}$= the relative error for particle size of *di*, $w_{o(di)}$ = the observed fall velocity of particle size of *di*, $w_{(di)}$ = the predicted fall velocity of particle size of *di* and *MRE* = the mean relative error. The mean relative error for the testing data, shown in table 2, is found as 4.8, indicating the good accuracy of the ANN model.

Table 3: The mean relative error of fall velocities estimated by various relations by using testing data

| Originator | References | MRE (%) |
|---|---|---|
| Rubey (1933) | [2] | 14.9 |
| Zanke (1977) | [5] | 10.6 |
| Hallermeier (1981) | [7] | 10.2 |
| Dietrich (1982) | [8] | 19.5 |
| Van Rijn (1989) | [9] | 10.8 |
| Concharov (1992) | [25] | 69.0 |
| Zhang (1993) | [27] | 7.0 |
| Zhu and Cheng (1993) | [28] | 8.9 |
| Julien (1995) | [10] | 13.8 |
| Soulsby (1997) | [26] | 11.0 |
| Cheng (1997) | [11] | 9.2 |
| Jimenez and Madsen (2003) | [12] | 6.2 |
| She *et al.* (2005) | [14] | 12.9 |
| Wu and Wang (2006) | [15] | 6.3 |
| ANN (This work) | | 4.8 |

Furthermore, using the testing data sets, the accuracy of ANN model for predicting the fall velocity of different particle sizes was compared with fourteen existing formulas listed in Table 3, included the recently formula presented by Wu and Wang [15]. The testing data sets have a Corey shape factor of 0.7. Table 3 shows the mean relative errors of the fourteen compared formulas. It can be seen that the formula of Concharov [25] has significant error, while Dietrich's formula [8],

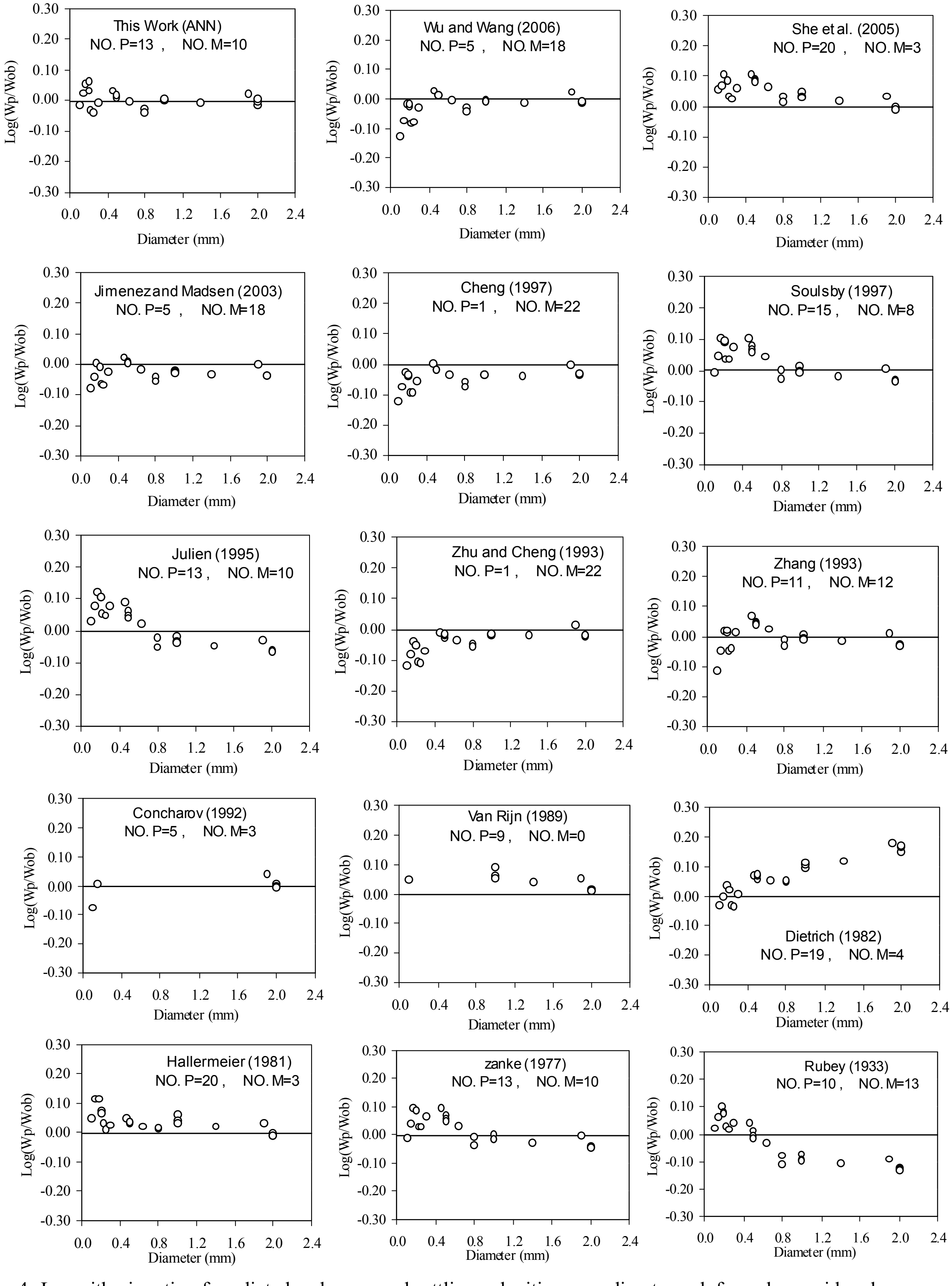


Fig. 4: Logarithmic ratio of predicted and measured settling velocities according to each formula considered NO.P= Number of overestimated data (Plus). NO.M= Number of underestimated data (Minus)

Rubey's formula [2], Julien's formula [10] have large errors. The five formulas of Jimenez and Madson [12], Wu and Wang [25], Zhang [28], Zhu and Cheng [30] and Cheng [4] perform well and have very close accuracies and the ANN model predicts better than all of them.

The comparison of all models is also shown in Fig. 4 which shows graphically the nature of the inaccuracy associated with each formula. The equivalence between number of overestimated (P) and underestimated (M) fall velocity predicted by ANN model, shows it's ability on estimating more accurate values for the amount of sediment fall velocity rather than other formulas mentioned in Table 3.

**Summary:** In this research, using 115 laboratorial date sets, an ANN model was developed and validated for predicting the settling velocity of individual natural sediment particles. Testing the ANN model using two sets of available data indicates a very strong agreement between the observed and predicted values. To compare with other formulas developed to estimate the particle fall velocity, 14 common equations were chosen. Computing the mean relative errors for all equations including recently proposed Wu and Wang formula [26] and ANN model, it is found that the new model has the smallest amount of MRE and hence it estimates most accurate values for sediment fall velocity.

## ACKNOWLEDGMENT

The writers thank Dr. J.A. Jimenez for his useful comments and his assistance to collect the relevant data.

## REFERENCES

1. Oseen, C., 1927. Hydrodynamik, Akademische Verlagsgesellschaft, Leipzig, Germany.
2. Rubey, W., 1933. Settling velocities of gravel, sand and silt particles. Am. J. Sci., 225: 325-338.
3. Rouse, H., 1938. Fluid mechanics for hydraulic engineers, Dover, New York.
4. Interagency Committee, 1957. Some fundamentals of particle size analysis: A study of methods used in measurement and analysis of sediment loads in streams. Rep. No. 12, Subcommittee on Sedimentation, Interagency Committee on Water Resources, St. Anthony Falls Hydraulic Laboratory, Minneapolis.
5. Zanke, U., 1977. Berechnung der sinkgeschwindigkeiten von sedimenten. Mitt. Des Franzius-Instituts fuer Wasserbau, Heft 46, Seite 243, Technical University, Hannover, Germany.
6. Yalin, M.S., 1977. Mechanics of sediment transport, Pergamon Press, Oxford, pp: 80-87.
7. Hallermeier, R.J., 1981. Terminal settling velocity of commonly occurring sand grains. Sedimentology, 28(6): 859-865.
8. Dietrich, W.E., 1982. Settling velocity of natural particles. Water Resource. Res., 18(6): 1615-1626.
9. Van Rijn, L.C., 1989. Handbook: Sediment transport by currents and waves. Rep. No. H 461, Delft Hydraulics, Delft, The Netherlands.
10. Julien, Y.P., 1995. Erosion and sedimentation, Cambridge University Press, Cambridge, U.K.
11. Cheng, N.S., 1997. Simplified settling velocity formula for sediment particle. J. Hyraul. Eng., ASCE, 123(8): 149-152.
12. Jimenez, J.A. and O.S. Madsen, 2003. A simple formula to estimate settling velocity of natural sediments. J. Waterw., Port, Coastal, Ocean Eng., 129(2): 70-78.
13. Brown, P.P. and D.F. Lawler, 2003. Sphere drag and settling velocity revisited. J. Environ. Eng., 129(3): 222-231.
14. She, K., L. Trim and D. Pope, 2005. Fall velocities of natural sediment particles: a simple mathematical presentation of the fall velocity law. J. Hydraul. Res., 43(2): 189-195.
15. Wu, W. and S.S.Y. Wang, 2006. Formulas for sediment porosity and settling velocity. J. Hydraul. Eng., 132(8): 858-862.
16. Alger, G.R. and D.B. Simons, 1968. Fall velocity of irregular shaped particles. J. Hydraul. Divi. Proc. ASCE, pp: 721-737.
17. Komar, P.D. and C.E. Reimers, 1978. Grain shape effects on settling rates. J. Geol., 86: 193-209.
18. Kohonen, T., 1988. An introduction to neural computing. Neural Networks, 1(1): 3-16.
19. Hagan, M.T., H.B. Demuth and M. Beale, 1996. Neural Network Design. PWS, Boston.
20. Lallahem, S., J. Mania, A. Hani and Y. Najjar, 2005. On the use of neural networks to evaluate groundwater levels in fractured media. J. Hydrol., 307: 92-111.
21. Hornik, K., M. Stinchcombe, H. White, 1989. Multilayer feedward networks are universal Approximators Neural Networks, 2: 359-366.

22. Engelund, F. and E. Hansen, 1972. A monograph on sediment transport in alluvial streams, 3rd Edn., Technical Press, Copenhagen, Denmark.
23. Fredsoe, J. and R. Deigaard, 1992. Mechanics of coastal sediment transport, World Scientific, Singapore.
24. Raudkivi, A.J., 1990. Loose boundary hydraulics, 3rd Edn., Pergamon Press, Oxford, U.K.
25. Ibad-zadeh, Y.A., 1992. Movement of sediment in open channels. S.P. Ghosh, translator, Russian translations series, Vol. 49, A.A. Balkema, Rotterdam, The Netherlands.
26. Soulsby, R.L., 1997. Dynamics of marine sands, Thomas Telford, London.
27. Yang, C.T., 1996. Sediment Transport-Theory and Practice. The McGraw-Hill Companies, Inc., pp: 6-10.
28. Zhu, L.J. and N.S. Cheng, 1993. Settlement of sediment particles. Res. Rep., Dept. Of River and Harbor Engrg., Nanjing Hydr. Res. Inst., Nanjing, China.
29. Baba, J. and D.B. Simons, 1981. Measurements and analysis of settling velocities of natural quartz sand grains. J. Sediment. Petrol., 51(2): 631-640.
30. Graf, W.H., 1971. Hydraulics of sediment transport, McGraw-Hill, New York.