

Research Article

Intelligent Flow Friction Estimation

Dejan Brkić¹ and Žarko Čojbašić²

¹European Commission, DG Joint Research Centre (JRC), Institute for Energy and Transport (IET), Energy Security, Systems and Market Unit, Via Enrico Fermi 2749, 21027 Ispra, Italy

²Faculty of Mechanical Engineering in Niš, University of Niš, Aleksandra Medvedeva 14, 18000 Niš, Serbia

Correspondence should be addressed to Dejan Brkić; dejanbrkic0611@gmail.com

Received 1 December 2015; Revised 5 February 2016; Accepted 7 February 2016

Academic Editor: Reinoud Maex

Copyright © 2016 D. Brkić and Ž. Čojbašić. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Nowadays, the Colebrook equation is used as a mostly accepted relation for the calculation of fluid flow friction factor. However, the Colebrook equation is implicit with respect to the friction factor (λ). In the present study, a noniterative approach using Artificial Neural Network (ANN) was developed to calculate the friction factor. To configure the ANN model, the input parameters of the Reynolds Number (Re) and the relative roughness of pipe (ϵ/D) were transformed to logarithmic scales. The 90,000 sets of data were fed to the ANN model involving three layers: input, hidden, and output layers with, 2, 50, and 1 neurons, respectively. This configuration was capable of predicting the values of friction factor in the Colebrook equation for any given values of the Reynolds number (Re) and the relative roughness (ϵ/D) ranging between 5000 and 10^8 and between 10^{-7} and 0.1, respectively. The proposed ANN demonstrates the relative error up to 0.07% which had the high accuracy compared with the vast majority of the precise explicit approximations of the Colebrook equation.

1. Introduction

To date, the Colebrook equation (1) is used as a mostly accepted standard for the calculation of fluid flow friction factor in pipes

$$\frac{1}{\sqrt{\lambda}} = -2 \cdot \log_{10} \left(\frac{2.51}{\text{Re} \cdot \sqrt{\lambda}} + \frac{\epsilon}{3.7 \cdot D} \right), \quad (1)$$

where λ is the Darcy friction factor (dimensionless); Re is Reynolds number (dimensionless), and ϵ/D is relative roughness of inner pipe surface (dimensionless).

The Colebrook equation is also somewhere known as the Colebrook-White equation or simply the CW equation [1]. Classifying the available data and those from experiment conducted in 1937 by himself and his professor White [2], Colebrook developed a curve fit which was describing transitional roughness, between the smooth and the rough turbulent zone [3]. The Colebrook equation is also considered as a proper base for the widely used Moody diagram with the exception of its laminar zone [4]. In other words, drawing his present famous diagram, Moody used Colebrook's equation for the whole turbulent zone and for the laminar zone

defined by $\lambda = 64/\text{Re}$. The Moody chart or Moody diagram is a graph in nondimensional form that relates the Darcy friction factor (λ), the Reynolds number (Re), and the relative roughness (ϵ/D) for fully developed flow in a circular pipe. It can be used to determine pressure drop or flow rate in such pipes. Although the accuracy of empirical equation of Colebrook can be disputable, it is sometimes essential to produce a fast, accurate, and robust resolution of this equation, which is particularly necessary for the scientific intensive computations and very often for comparisons [5]. Unfortunately, the Colebrook equation suffers from being implicit with respect to the friction factor (λ). It cannot be rearranged to derive the friction factor directly with no approximate calculation. Many different strategies are used to calculate or to estimate the friction factor accurately [1, 6–8].

There are a group of studies investigating the use of Artificial Neural Network (ANN) to estimate the friction factor. For instance, the intelligent estimation of hydraulic resistance for Newtonian fluids has been investigated in some of recent studies [9–13]. For the other types of fluids used in agriculture, food engineering, petroleum engineering, and so forth, such as power-law, Bingham, Herschel-Bulkley,

and other types of non-Newtonian fluids, the shown ANN cannot be used in the most cases. However, the developed methodology for training can be used with appropriate dataset or appropriate equations to produce relevant solution in such cases where the aforementioned ANN cannot be used [14–16]. Application of ANN for simulation of other types of friction factor rather than Colebrook, namely, Hazen–Williams friction coefficient for small-diameter polyethylene pipes, can also be found in the literature [17], while more recently other attempts of ANN usage for modeling friction factors in pipes have been reported [18, 19].

Nowadays, not only can the ANN approach be used in hydraulics and for simulation of fluid flow, but also it can be widely applied in the various branches of engineering, such as for the control systems [19, 20], as an auxiliary tool in medicine [21–25], a flow pattern indicator for gas-liquid flow in a microchannel [26], and an extension of structural mechanics tools for fast determination of structural response [27]. Also combined neurofuzzy systems (NFS) approach can be used for different purposes such as student modeling system, medical system, economic system, electrical and electronics system, traffic control, image processing and feature extraction, manufacturing and system modeling, forecasting and predictions, and social sciences [28].

2. Definition of the Problem

In the present study, in order to produce an efficient and accurate procedure for estimation of the flow friction factor (λ), an approach based on the computationally intelligent system was used. The Artificial Neural Network (ANN) for the solution of the problem is developed. The ANN models like the one shown here can be easily generated in the MATLAB software.

First, the raw datasets calculated using the Colebrook equation were used to train the ANN model and then the unknown friction factors (λ) were predicted by obtaining the ANN structure with a low relative error. In this paper, the empirical Colebrook equation (1) and its accurate iterative solution will be treated as “accurate by the default” or “absolutely accurate” (sign “=” is used, while for the approximations listed in Appendix sign “ \approx ” is used).

Hydraulic resistance depends on the flow rate which is considered as the main problem in determination of the hydraulic flow friction factor (λ). For a pipe, the hydraulic resistance usually is expressed through the Darcy friction factor (λ) which is not a constant quantity. Friction factor (λ) is related to the flow rate or more precisely to the Reynolds number (Re) and the relative roughness (ϵ/D). In addition, both of them, the Reynolds number (Re) and the relative roughness (ϵ/D), are dependent on the flow rate. In fact, the Reynolds number (Re) is affected by flow velocity while the relative roughness (ϵ/D) depends on the thickness of a region of flow inside pipes, termed as boundary layer, which occurs closely to the inner surface of pipe wall [29, 30]. On the contrary, in this paper the relative roughness (ϵ/D) retains its classical definition, which implies it should not vary with the flow rate (it will be treated effectively as a geometric quantity and thus should be constant regardless of flow rate with the

caveat that the flow is turbulent). Furthermore, it is obvious that changes of the hydraulic resistance in the turbulent zone are governed by the nonlinear law. In general, these hydraulic resistances in turbulent zone can be modeled as logarithmic-law or power-law [31]. The Colebrook equation belongs to the logarithmic-law.

As it was mentioned, the main problem of the Colebrook equation is related to its implicit form with respect to the friction factor (λ) which cannot be evaluated without the approximate calculation (the Colebrook equation is a transcendental function). Therefore, different strategies are used to find adequate solution for Colebrook equation: iterative solution (in the present study, it was assumed that values calculated by this method are highly accurate) [6, 7], use of plenty of available explicit approximations of the Colebrook equation derived by numerous mathematical or numerical approaches [6, 8, 32, 33], using some graphical interpretations such as the Moody diagram [4], and so forth.

It should be taken into account that the Moody diagram cannot be used as a reliable and accurate replacement for the Colebrook equation as its reading error can be even more than few percent [10, 34, 35]. Using iterative methods, namely, the Newton-Raphson, the friction factor (λ) can be calculated from the Colebrook equation with high accuracy where the convergence of 0.01% requires less than 7 iterations. This accuracy (0.01%) should not be confused with the accuracy of the explicit approximations of the Colebrook equation [36]. Reviewing the relevant literature, one can realize that the vast majority of these approximations are extremely accurate and they can be used instead of implicit Colebrook equation to calculate the friction factor (λ). However, the final maximal error caused by approximation should be estimated as the sum of the real maximal error of certain approximation and the error caused by iterative procedure.

The two most accurate explicit approximations with the relative errors up to 0.0026% and 0.0083% are those implied by Čojbašić and Brkić [37]. Moreover, there are plenty of other approximations with the relative errors above 0.13% [6]. Indeed, use of the highly accurate approximations could complicate the fluid flow calculations. However, use of the advanced and powerful computers and codes can partially solve this problem and reduce the computational burden [38].

In this study, the implied ANN structure led to a low relative error compared to the accurate iterative solution. In addition, the computational burden used to run the applied ANN structure was equal or lower than that of explicit approximations, and it, especially, was less than that of the iterative solution of the original Colebrook equation, while the accuracy of the ANN approach remains significantly high.

3. Methodology

3.1. Preparation of the Dataset. In order to generate the training set for the ANN model, the Colebrook equation was solved iteratively. The iterative solution is used because the highly accurate solution of the friction factor (λ) was required, while in the meantime the computational burden was irrelevant since it was a onetime effort to prepare the training data. The training dataset can be efficiently prepared

using the spreadsheet solvers, such as MS Excel which is used in the particular case presented here [6, 7]. In order to obtain the highest accuracy in the calculation using MS Excel, the iterative calculation should be enabled and the maximum number of iterations (it is set to 32,767 iterations which was the maximum number of cycles allowed by the software with the highest precision) has to be set [7].

In order to train the presented ANN model, input dataset (Electronic Appendix A: MS Excel spreadsheet with the set of 90 thousand combinations used for training of the Artificial Neural Network (ANN) (see Supplementary Material available online at <http://dx.doi.org/10.1155/2016/5242596>) involving 90,000 triplets was used in which the values of the Darcy friction factor (λ) were generated using values of the Reynolds number (Re) and the relative roughness (ϵ/D) ranged $5000-10^8$ and $10^{-7}-0.1$, respectively. In order to use input datasets, the values of the Reynolds number (Re) and the relative roughness (ϵ/D) had to be normalized. The used approach will be comprehensively explained in the next parts.

3.2. Structure and Training of the ANN. The feedforward neural network structure which consists of three layers is used (Figure 1). The first, input layer has two neurons, the second, hidden layer has fifty neurons, and the third, output layer has one neuron, with a sigmoid transfer function in the hidden layer and a linear transfer function in the output layer.

In general, an ANN should be trained, or adapted, either before or during its use. The used ANN network was properly trained and validated by supervised offline training prior to network application in which the data obtained by the iterative solution of the Colebrook equation were applied.

Almost every neural network consists of a large number of simple processing elements that are variously called neurons, nodes, cells, or units, connected to other neurons by means of direct communication links, each with an associated weight and bias. The weights represent information being used by the net to produce output for given inputs. The most common feedforward net has two or more layers of processing units in the adjacent layers. Generally speaking, ANN is able to efficiently imitate functions and recognize patterns. They can be trained to solve a problem (ability to learn). The quality of this solution heavily depends on the quantity of available data for training and the structure of a network.

It should be underlined that the developed ANN (the generated ANN is attached as Electronic Appendix B to this paper; file ColebrookANN.mat) does not use the Colebrook equation for the calculation. It uses only the results produced by the Colebrook equation to establish its inner patterns. Every neural network is considered as a “Black box” system; therefore, it can be viewed in terms of its inputs and outputs without any knowledge about its internal working and inner components.

However, the main issue of the present network is related to the ranges of input parameter in which the relative roughness (ϵ/D) is extremely small as it ranged from 10^{-7} to 0.1, while another parameter, the Reynolds number (Re), is considerably large in the range of 2320 to 10^8 . This problem can prevent the ANN from being properly trained and it

will lead to the less accurate results in application phase. Therefore, the raw input dataset should be normalized to provide the input data for the ANN with the approximately same order of magnitude.

In order to address this issue, the logarithmic transformation can be done where the Reynolds number (Re) and the relative roughness (ϵ/D) were replaced by $\log(\text{Re})$ and $-\log(\epsilon/D)$, respectively. These transformations translated (copied) input values into the new domain where $\log(\text{Re})$ is in range between 3.7 and 8 and $-\log(\epsilon/D)$ is in range between 1 and 6.5. Dataset set with the 90,000 combinations of the Reynolds number (Re), the relative roughness (ϵ/D), and related friction factor (λ) was prepared in MS Excel as already explained. Full prepared dataset was divided into training, validation, and testing subsets:

- (i) The training sample (70%, 63,000 triplets) was presented to the ANN during the training,
- (ii) the validation sample (15%, 10,500 triplets) was used to measure generalization of the ANN, that is, to stop the training when the generalization does not improve anymore (i.e., this prevents the so-called “overfitting”),
- (iii) the testing sample (15%, 10,500 triplets) had no effect on the training and so it provided an independent measure of performance of the ANN during and after training.

Inputs were normalized and used for the training of the ANN which is indicated in Figure 1. The concept of the training process is shown in Figure 2. The Neural Network Toolbox of MATLAB software was used to simulate the proposed ANN for the shown flow friction problem.

3.3. Use of the ANN. When the training process with 90,000 inputs/output combinations of data was finalized, the generated ANN was saved under the name of “ColebrookANN” for later uses. In such a way, the ANN can be further used for the accurate estimation of the flow friction factor (λ). The Colebrook equation was used for the training process of the ANN model. Then, the generated ANN will use inputs and produce results that follow this pattern from the learning phase for any unknown combination of inputs. The phase of exploitation of network is shown in Figure 3.

For the presented ANN, the process of training lasted few hours. Afterwards, the ANN can be used to estimate flow friction factor (λ), accurately. This can be carried out using MATLAB software by loading network previously saved with the name “ColebrookANN” using command:

```
load ColebrookANN.mat
```

The hydraulic friction factor (λ) can be evaluated using single line in MATLAB:

```
lambda=sim(ColebrookANN, [log10(Re); -log10(RPR)]),
```

where Re denotes the values for the Reynolds number (Re) while RPR denotes relative roughness (ϵ/D), that is, Relative Pipe Roughness (RPR), in order to avoid Greek letters

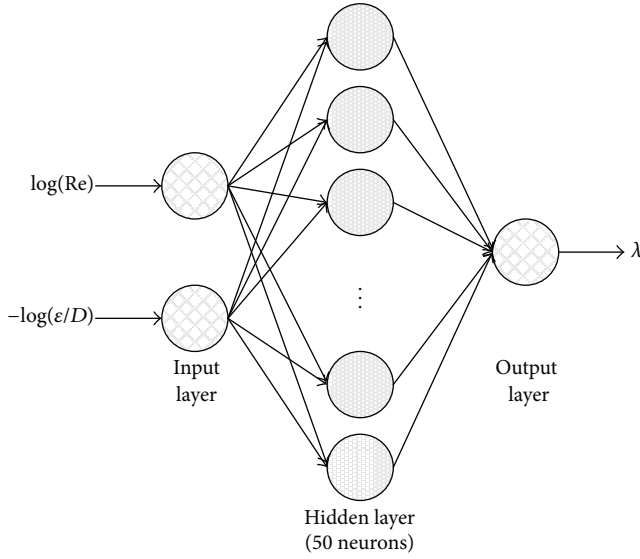


FIGURE 1: Structure of the proposed ANN.

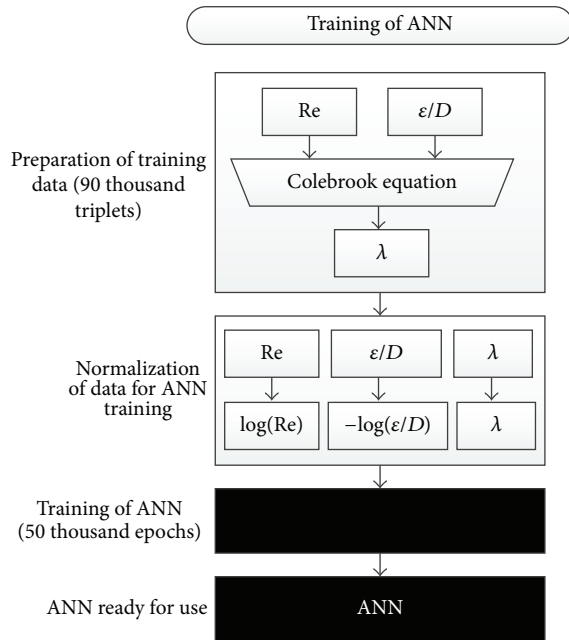


FIGURE 2: The scheme of training process of the ANN.

in the code. Due to MATLAB exquisite matrix handling capabilities, the sets of pairs of input data can be prepared in one row by multiple columns vector variables of the Re and the RPR. In this case the MATLAB produces vector lambda involving the calculated friction factors (λ) for each input data pair in fraction of time, even for the large datasets.

In order to determine the hydraulic friction factor (λ) using ANN, the sufficiently large training dataset was used which was in contrast to other published results in this field [9–13]. The proposed network can outperform even the most accurate approximations to the Colebrook equation.

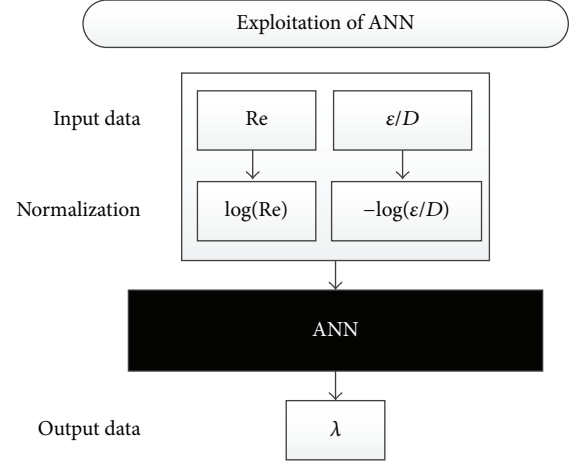


FIGURE 3: Exploitation of the ANN.

4. Results and Discussion

4.1. Model Performance. In order to examine the performance of a model, approximation quality, model complexity, and model interpretability should be addressed. In fact, the approximation/prediction error is often used as an assessment criterion. There are different criteria in the literature to assess the model performance. It is possible that the worst case or the average deviation is crucial [39, 40].

For training of the presented ANN, the back propagation Levenberg-Marquardt algorithm was used, while the Mean Squared Error (MSE) was used as performance measure during the training phase. The values of MSE for this ANN structure were calculated to be 10^{-12} after 5,000 epochs of training (Figure 4). The main goal was to minimize the performance function, in this case MSE function, which is defined as

$$\text{MSE} = \frac{1}{n} \sum_{k=1}^n e_k^2 = \frac{1}{n} \sum_{k=1}^n (t_k - y_k)^2, \quad (2)$$

where n denotes number of samples, e_k denotes neural network error, and t_k denotes target values, while y_k are network output values. The training algorithm used in all cases was Levenberg-Marquardt algorithm [41], where network weights w are updated by the equation $\mathbf{w}_{k+1} = \mathbf{w}_k - (\mathbf{J}_k^T \mathbf{J}_k + \mu \mathbf{I})^{-1} \mathbf{J}_k e_k$ and which is based on the approximation of Hessian matrix $\mathbf{H} = \mathbf{J} \mathbf{J}^T + \mu \mathbf{I}$, where \mathbf{J} denotes Jacobian matrix, \mathbf{I} denotes identity matrix, and μ is always positive so-called combination coefficient. The Levenberg-Marquardt algorithm was selected as being stable, fast, and reliable.

The training of the proposed ANN structure was done through 5,000 epochs. The Mean Squared Error (MSE) of this ANN structure was calculated to be 10^{-12} after which there was no further tendency to decrease. In addition, the same results were obtained with the tested ANN structures involving 100 neurons in a hidden layer and with the two hidden layers containing 50 neurons in each of them. However, the tested ANN structure with 30 neurons in one hidden layer

TABLE 1: Relative error of friction factor produced by the shown ANN over the practical domain of the relative roughness (ϵ/D) and the Reynolds number (Re).

Relative error (%) Reynolds number (Re)	Relative roughness (ϵ/D)									
	10^{-6}	$5 \cdot 10^{-6}$	10^{-5}	$5 \cdot 10^{-5}$	10^{-4}	$5 \cdot 10^{-4}$	10^{-3}	$5 \cdot 10^{-3}$	10^{-2}	$5 \cdot 10^{-2}$
10^4	0.00134	0.00088	0.00031	0.00017	0.00123	0.00141	0.00041	0.00099	0.00096	0.00069
$5 \cdot 10^4$	0.00102	0.00174	0.00080	0.00096	0.00220	0.00163	0.00247	0.00063	0.00224	0.00124
10^5	0.00114	0.00145	0.00125	0.00356	0.00099	0.00384	0.00097	0.00117	0.00104	0.00076
$5 \cdot 10^5$	0.00181	0.00032	0.00287	0.00084	0.00047	0.00090	0.00028	0.00011	0.00055	0.00064
10^6	0.00163	0.00246	0.00126	0.00073	0.00419	0.00440	0.00176	0.00190	0.00023	0.00053
$5 \cdot 10^6$	0.00449	0.00672	0.00207	0.00377	0.00012	0.00077	0.00071	0.00031	0.00038	0.00074
10^7	0.00126	0.00054	0.00417	0.00527	0.00005	0.00089	0.00015	0.00033	0.00063	0.00186
$5 \cdot 10^7$	0.01946	0.00382	0.00490	0.00835	0.00260	0.00174	0.00011	0.00071	0.00038	0.00022
10^8	0.06060	0.05266	0.03614	0.02413	0.01682	0.00410	0.00165	0.00544	0.00579	0.00068

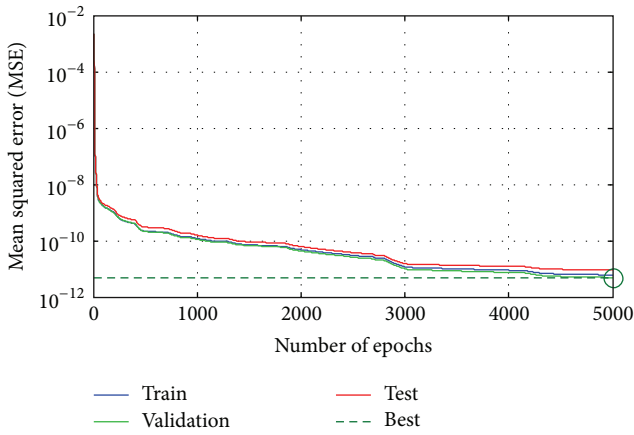


FIGURE 4: The Mean Squared Error (MSE) during the process of training of the proposed ANN.

resulted in a lower accuracy in comparison with the former tested structures, even after 10,000 epochs of training.

4.2. Accuracy of the Estimated Results. For the purpose of comparison, it is better to use the relative error than the Mean Squared Error (MSE) which was used during the training process of the proposed ANN. The maximum relative error of the proposed feedforward ANN structure, with one hidden layer containing 50 neurons, compared with the iterative solution of the Colebrook equation, was up to 0.07% (Table 1).

It should be taken into account that there are three levels of the accuracy [36, 41]:

- (1) The first level is related to the nature of the Colebrook equation which is an empirical relation (in fact, there is a possibility of using other equations with higher accuracy, and accordingly the showed methodology can be used in order to develop the appropriate ANN for such a case).
- (2) The second level explains the accuracy related to the solution of the Colebrook equation; the Colebrook equation can be solved precisely using the iterative procedure (in this paper, the term “accurate by

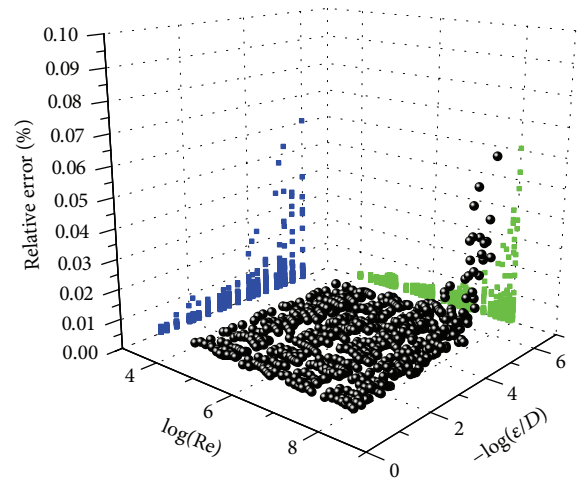


FIGURE 5: Distribution of the estimated error produced by the ANN compared with the Colebrook equation in normalized domain which is suitable for training of the ANN (verification in MATLAB).

default” or “absolutely accurate” and the related error can be neglected in many cases).

- (3) The third one is related to the proposed ANN structures and relevant approximations which can be used to avoid iterative procedure; their errors can be estimated and compared with the error of iterative solution (obtained error of the suggested ANN structure belongs to the third category).

The relative error of friction factor estimated through the proposed ANN structure in this is up to 0.07% (Figures 5 and 6). This means that proposed ANN approach can be used not only as extremely accurate approach, but also as a computationally effective one.

Furthermore, to some extent, an increase in the complexity of the ANN structure would augment its potential to produce even more accurate results. Hence, the right balance of accuracy and complexity is necessary during the network design phase. Additionally, accuracy depends on the quantity of terms in the training set. The complexity of network in the phase of exploitation is relatively unimportant since the ANN

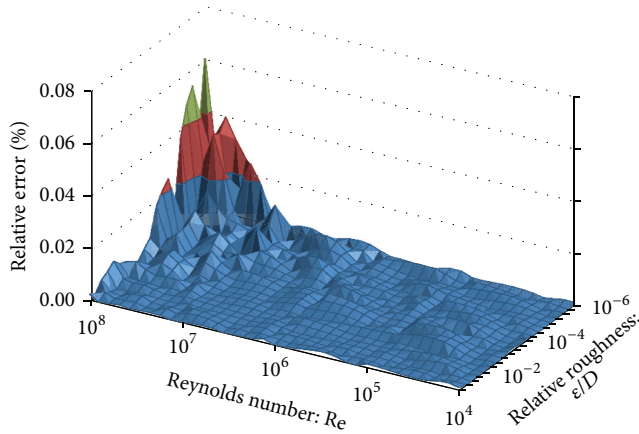


FIGURE 6: Distribution of the estimated error produced by the ANN compared with the Colebrook equation (verification in MS Excel).

is a sort of “black box.” It can produce outputs for inputs and its inner complexity is not crucial [47, 48].

Users would easily apply the ANN without any difficulty due to its structure complexity, in contrast to use of the approximate formulas [38]. The same circumstances of comfort can be experienced by users applying the prepared computer codes for the approximate formulas. Users will be able to enter input data into a program and a computer should be able further to produce outputs without any inconvenience.

According to Figures 5 and 6, the relative error is not equally distributed over the entire practical range of the Reynolds number (Re) and the relative roughness (ϵ/D). The same situation with this distribution of the error would occur for the explicit approximations as shown by Brkić [6, 7] and Winning and Coole [33]. The relative error produced by the ANN is accumulated in the zone with small values of the relative roughness (ϵ/D) and the high values of the Reynolds number (Re). The distribution of the relative error is also shown in Table 1. According to Table 1 the maximum relative error was calculated to be 0.0606% for $Re = 10^8$ and $\epsilon/D = 10^{-6}$.

4.3. Comparative Analysis. Having looked at the existing approximations of Colebrook equation [6, 7], one can obviously realize that the available explicit approximations of the Colebrook equation are either inaccurately simple or intricately accurate. In fact, the complexity of explicit approximations (e.g., approach with the Lambert W -function [8, 49]) was considered as a serious issue few decades ago when pocket calculators were widely used [38]. Nowadays, even the very complex approximations can be easily used in computer codes. In the study conducted by Brkić [6], it was concluded that the five most available approximations from the literature had the maximum relative error up to 0.15%. These approximations were suggested by Zigrang and Sylvester [46], Serghides [42], Romeo et al. [43], Buzzelli [45], and Vatankhah and Kouchakzadeh [44] (even more accurate approximations are shown in Vatankhah [50] where their accuracy is comparable with accuracy of approximations shown in Čojbašić and Brkić [37]). Furthermore, Čojbašić

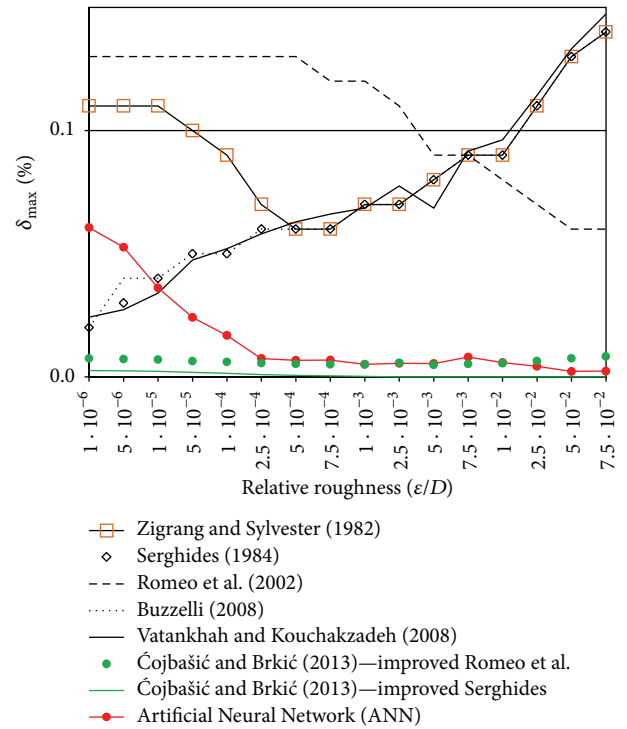


FIGURE 7: Maximal relative error produced by ANN compared with the seven most accurate explicit approximations of Colebrook equation where ϵ/D is used for x -axis.

and Brkić [37] applied genetic algorithm optimization technique (also genetic technique are used in [51, 52]). This technique improved two of these accurate approximations suggested by Serghides [42] and Romeo et al. [43] to reach even extreme level of accuracy with the relative error up to 0.0026% and 0.0083%, respectively. All mentioned explicit approximations are listed in Appendix of this paper (they are also attached to this paper as Electronic Appendix C (PDF file with all approximations of the Colebrook equation mentioned through text with their MATLAB codes and MS Excel codes)). The accuracy of the proposed ANN in the present work was compared with accuracy of these approximations which is shown in Figure 7 where relative roughness (ϵ/D) is used as the base for the x -axis of the diagram. Moreover, in Table 2, the Reynolds number (Re) is used as the base. This means that, in the case of using relative error of the presented ANN from Figure 7, the maximum value of the relative error can be chosen from each column of Table 1, while, in the case of using of Table 2, the maximum value of the relative error can be chosen from each row of Table 1.

The results of comparative analysis which were reported in Figure 7 revealed that the implied ANN structure could outperform the vast majority of the most accurate approximations in the large area of data domain. In addition, the suggested ANN structure in this study might be used with the most accurate explicit approximations of the Colebrook equation implied by Čojbašić and Brkić [37], Romeo et al. [43], Buzzelli [45], Serghides [42], Zigrang and

TABLE 2: Maximal relative error produced by the ANN compared with the seven most accurate explicit approximations of Colebrook equation; the Reynolds number (Re) is used as the base.

Reynolds number (Re)	Maximal relative error (%)							
	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)
10^4	0.00141	0.00074	0.00569	0.12272	0.13563	0.13453	0.13301	0.13313
$5 \cdot 10^4$	0.00247	0.00219	0.00574	0.14112	0.13784	0.11047	0.13736	0.13736
10^5	0.00384	0.00246	0.00698	0.14467	0.13812	0.10281	0.13793	0.13793
$5 \cdot 10^5$	0.00287	0.00250	0.00802	0.14712	0.13841	0.08915	0.13839	0.13839
10^6	0.00440	0.00235	0.00816	0.14727	0.13846	0.08426	0.13845	0.13845
$5 \cdot 10^6$	0.00672	0.00167	0.00826	0.14725	0.13850	0.07315	0.13850	0.13850
10^7	0.00527	0.00122	0.00828	0.14722	0.13851	0.06754	0.13850	0.13850
$5 \cdot 10^7$	0.01946	0.00022	0.00829	0.14718	0.13851	0.04876	0.13851	0.13851
10^8	0.06060	0.00005	0.00829	0.14718	0.13851	0.04841	0.13851	0.13851

(a)-Artificial Neural Network (ANN).

(b)-Ćojbašić and Brkić [37]-Improved Serghides [42]; (A.7).

(c)-Ćojbašić and Brkić [37]-Improved Romeo et al. [43]; (A.6).

(d)-Vatankhah and Kouchakzadeh [44]; (A.2).

(e)-Buzzelli [45]; (A.1).

(f)-Romeo et al. [43]; (A.3).

(g)-Serghides [42]; (A.4).

(h)-Zigrang and Sylvester [46]; (A.5).

Sylvester [46], and Vatankhah and Kouchakzadeh [44]. The maximum relative errors for these approximations were evaluated to be 0.0026%, 0.13%, 0.14%, 0.14%, 0.14%, and 0.15%, respectively.

5. Conclusion

In order to evaluate the friction factor, the sophisticated ANN model was developed. The model includes three layers of input, hidden, and output neurons with 2, 50, and 1 neurons, respectively. The trained ANN is able to predict friction factor (λ) with the relative error of less than 0.07%. Based on the performed comparative analysis, the developed ANN produces the lowest relative error in comparison with most of accurate explicit approximations of the Colebrook equation. Furthermore, to deal with the low accuracy of the Colebrook equation or to facilitate for specific needs, the suggested ANN structure could be trained using some of the other available precise approximations or experimental data [53, 54] (although each new training will produce different inner pattern among neurons [55], the final estimation of friction factor will remain with almost the same level of accuracy) and even using combination of these for different parts of input domains which could be considered as significant advantage [56]. For these reasons, this suggested ANN structure in the present study would be worthwhile to solve flow problems involving repetitive calculations of the friction factor (λ). An important disadvantage might be the fact that significant number of training patterns is required to obtain accuracy level presented in this paper, but this would be with limited impact since the problem can be overwhelmed with onetime effort.

In our approach we tried to keep the solution simple and provide single neural network that covers the whole range of inputs, but further interesting research direction would be to design several networks covering parts of input spaces

and working in conjunction possibly providing improved accuracy and sacrificing simplicity of the solution. Also, following our own results and results of others regarding application of other techniques of computational intelligence for the same problem, the ANN presented here could potentially be cross-fertilized with them in an attempt to improve results, where primarily genetic optimization of the network structure might be promising.

Appendix

Approximations of the Colebrook equation for flow friction used in this paper are as follows (MATLAB and MS Excel codes for the shown approximations are listed in Electronic Appendix C of this paper) [λ , Re, ε/D are with the same meaning as in (1) of this paper while A_{1-16} are auxiliary terms]:

(i) Buzzelli approximation [45]:

$$\frac{1}{\sqrt{\lambda}} \approx A_1 - \left(\frac{A_1 + 2 \cdot \log_{10}(A_2/\text{Re})}{1 + 2.18/A_2} \right),$$

$$A_1 \approx \frac{(0.774 \cdot \ln(\text{Re})) - 1.41}{(1 + 1.32 \cdot \sqrt{\varepsilon/D})}, \quad (\text{A.1})$$

$$A_2 \approx \frac{1}{3.7} \cdot \frac{\varepsilon}{D} \cdot \text{Re} + 2.51 \cdot A_1.$$

(ii) Vatankhah and Kouchakzadeh [44, 50, 57]:

$$\frac{1}{\sqrt{\lambda}} \approx 0.8686 \cdot \ln \left(\frac{0.4587 \cdot \text{Re}}{(A_3 - 0.31)^{A_4}} \right),$$

$$A_3 \approx 0.124 \cdot \text{Re} \cdot \frac{\varepsilon}{D} + \ln(0.4587 \cdot \text{Re}), \quad (\text{A.2})$$

$$A_4 \approx \frac{A_3}{A_3 + 0.9633}.$$

(iii) Romeo et al. approximation [43]:

$$\begin{aligned}\frac{1}{\sqrt{\lambda}} &\approx -2 \cdot \log_{10} \left(\frac{1}{3.7065} \cdot \frac{\varepsilon}{D} - \frac{5.0272}{\text{Re}} \cdot A_5 \right), \\ A_5 &\approx \log_{10} \left(\frac{1}{3.827} \cdot \frac{\varepsilon}{D} - \frac{4.567}{\text{Re}} \cdot A_6 \right), \\ A_6 &\approx \log_{10} \left(\left(\frac{1}{7.7918} \cdot \frac{\varepsilon}{D} \right)^{0.9924} \right. \\ &\quad \left. + \left(\frac{5.3326}{208.815 + \text{Re}} \right)^{0.9345} \right).\end{aligned}\tag{A.3}$$

(iv) Serghides approximation [42]:

$$\begin{aligned}\frac{1}{\sqrt{\lambda}} &\approx A_7 - \frac{(A_8 - A_7)^2}{A_9 - 2 \cdot A_8 + A_7}, \\ A_7 &\approx -2 \cdot \log_{10} \left(\frac{1}{3.7} \cdot \frac{\varepsilon}{D} + \frac{12}{\text{Re}} \right), \\ A_8 &\approx -2 \cdot \log_{10} \left(\frac{1}{3.7} \cdot \frac{\varepsilon}{D} + \frac{2.51 \cdot A_7}{\text{Re}} \right), \\ A_9 &\approx -2 \cdot \log_{10} \left(\frac{1}{3.7} \cdot \frac{\varepsilon}{D} + \frac{2.51 \cdot A_8}{\text{Re}} \right).\end{aligned}\tag{A.4}$$

(v) Zigrang and Sylvester approximation [46, 58]:

$$\begin{aligned}\frac{1}{\sqrt{\lambda}} &\approx -2 \cdot \log_{10} \left(\frac{1}{3.7} \cdot \frac{\varepsilon}{D} - \frac{5.02}{\text{Re}} \cdot A_{10} \right), \\ A_{10} &\approx \log_{10} \left(\frac{1}{3.7} \cdot \frac{\varepsilon}{D} - \frac{5.02}{\text{Re}} \cdot A_{11} \right), \\ A_{11} &\approx \log_{10} \left(\frac{1}{3.7} \cdot \frac{\varepsilon}{D} + \frac{13}{\text{Re}} \right).\end{aligned}\tag{A.5}$$

(vi) Čojbašić and Brkić approximation [37, 43]:

$$\begin{aligned}\frac{1}{\sqrt{\lambda}} &\approx -2 \cdot \log_{10} \left(\frac{1}{3.7106} \cdot \frac{\varepsilon}{D} - \frac{5}{\text{Re}} \cdot A_{12} \right), \\ A_{12} &\approx \log_{10} \left(\frac{1}{3.8597} \cdot \frac{\varepsilon}{D} - \frac{4.795}{\text{Re}} \cdot A_{13} \right), \\ A_{13} &\approx \log_{10} \left(\left(\frac{1}{7.646} \cdot \frac{\varepsilon}{D} \right)^{0.9685} \right. \\ &\quad \left. + \left(\frac{4.9755}{206.2795 + \text{Re}} \right)^{0.8759} \right).\end{aligned}\tag{A.6}$$

(vii) Čojbašić and Brkić approximation [37, 42]:

$$\begin{aligned}\frac{1}{\sqrt{\lambda}} &\approx A_{14} - \frac{(A_{15} - A_{14})^2}{A_{16} - 2 \cdot A_{15} + A_{14}}, \\ A_{14} &\approx -2 \cdot \log_{10} \left(\frac{1}{3.71} \cdot \frac{\varepsilon}{D} + \frac{12.585}{\text{Re}} \right), \\ A_{15} &\approx -2 \cdot \log_{10} \left(\frac{1}{3.71} \cdot \frac{\varepsilon}{D} + \frac{2.51 \cdot A_{14}}{\text{Re}} \right), \\ A_{16} &\approx -2 \cdot \log_{10} \left(\frac{1}{3.71} \cdot \frac{\varepsilon}{D} + \frac{2.51 \cdot A_{15}}{\text{Re}} \right).\end{aligned}\tag{A.7}$$

Additional Points

Software packages used for this research are MS Excel ver. 2007 and MATLAB R2010a by MathWorks. The paper is registered in the internal system for publication PUBSY of the Joint Research Centre of the European Commission under no. JRC100455.

Disclosure

The views expressed are purely those of the authors and may in any circumstance be regarded as stating an official position of neither the European Commission nor the University of Niš.

Competing Interests

The authors declare that there are no competing interests regarding the publication of this paper.

Acknowledgments

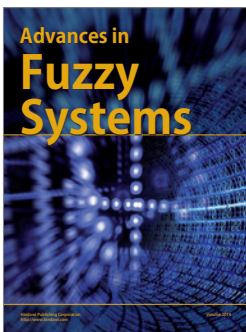
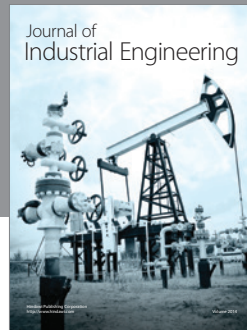
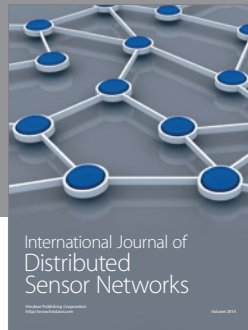
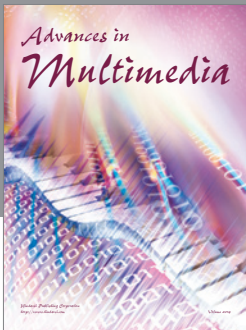
The work of Žarko Čojbašić has been supported by the Ministry of Education, Science and Technological Development of the Republic of Serbia under Grant TR35016.

References

- [1] D. Brkić, “W solutions of the CW equation for flow friction,” *Applied Mathematics Letters*, vol. 24, no. 8, pp. 1379–1383, 2011.
- [2] C. F. Colebrook and C. M. White, “Experiments with fluid friction in roughened pipes,” *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 161, no. 906, pp. 367–381, 1937.
- [3] C. F. Colebrook, “Turbulent flow in pipes, with particular reference to the transition region between the smooth and rough pipe laws,” *Journal of the Institution of Civil Engineers*, vol. 11, no. 4, pp. 133–156, 1939.
- [4] L. F. Moody, “Friction factors for pipe flow,” *Transaction of ASME*, vol. 66, no. 8, pp. 671–684, 1944.
- [5] D. Clamond, “Efficient resolution of the colebrook equation,” *Industrial and Engineering Chemistry Research*, vol. 48, no. 7, pp. 3665–3671, 2009.
- [6] D. Brkić, “Review of explicit approximations to the Colebrook relation for flow friction,” *Journal of Petroleum Science and Engineering*, vol. 77, no. 1, pp. 34–48, 2011.
- [7] D. Brkić, “Determining friction factors in turbulent pipe flow,” *Chemical Engineering*, vol. 119, no. 3, pp. 34–39, 2012.
- [8] D. Brkić, “Comparison of the Lambert W-function based solutions to the Colebrook equation,” *Engineering Computations*, vol. 29, no. 6, pp. 617–630, 2012.
- [9] M. Özger and G. Yildirim, “Determining turbulent flow friction coefficient using adaptive neuro-fuzzy computing technique,” *Advances in Engineering Software*, vol. 40, no. 4, pp. 281–287, 2009.
- [10] G. Yildirim, “Computer-based analysis of explicit approximations to the implicit Colebrook-white equation in turbulent flow friction factor calculation,” *Advances in Engineering Software*, vol. 40, no. 11, pp. 1183–1190, 2009.

- [11] W. H. Shayya and S. S. Sablani, "An artificial neural network for non-iterative calculation of the friction factor in pipeline flow," *Computers and Electronics in Agriculture*, vol. 21, no. 3, pp. 219–228, 1998.
- [12] F. Salmasi, R. Khatibi, and M. A. Ghorbani, "A study of friction factor formulation in pipes using artificial intelligence techniques and explicit equations," *Turkish Journal of Engineering and Environmental Sciences*, vol. 36, no. 2, pp. 121–138, 2012.
- [13] D. A. Fadare and U. I. Ofidhe, "Artificial neural network model for prediction of friction factor in pipe flow," *Journal of Applied Sciences Research*, vol. 5, no. 6, pp. 662–670, 2009.
- [14] S. S. Sablani and W. H. Shayya, "Neural network based non-iterative calculation of the friction factor for power law fluids," *Journal of Food Engineering*, vol. 57, no. 4, pp. 327–335, 2003.
- [15] S. S. Sablani, W. H. Shayya, and A. Kacimov, "Explicit calculation of the friction factor in pipeline flow of Bingham plastic fluids: a neural network approach," *Chemical Engineering Science*, vol. 58, no. 1, pp. 99–106, 2003.
- [16] W. H. Shayya, S. S. Sablani, and A. Campo, "Explicit calculation of the friction factor for Non-Newtonian fluids using artificial neural networks," *Developments in Chemical Engineering and Mineral Processing*, vol. 13, no. 1-2, pp. 5–20, 2005.
- [17] G. Yildirim and M. Özger, "Neuro-fuzzy approach in estimating Hazen–Williams friction coefficient for small-diameter polyethylene pipes," *Advances in Engineering Software*, vol. 40, no. 8, pp. 593–599, 2009.
- [18] M. Yazdi and A. Bardi, "Estimation of friction factor in pipe flow using artificial neural networks," *Canadian Journal on Automation, Control & Intelligent Systems*, vol. 2, no. 4, pp. 52–56, 2011, https://www.researchgate.net/profile/Mehran_Yazdi/publication/265939967_Estimation_of_Friction_Factor_in_Pipe_Flow_Using_Artificial_Neural_Networks/links/551a8fd90cf26cbb81a2f1bc.pdf.
- [19] S. M. Besarati, P. D. Myers, D. C. Covey, and A. Jamali, "Modeling friction factor in pipeline flow using a GMDH-type neural network," *Cogent Engineering*, vol. 2, no. 1, p. 1056929, 2015.
- [20] D. Brkić and T. I. Tanasković, "Systematic approach to natural gas usage for domestic heating in urban areas," *Energy*, vol. 33, no. 12, pp. 1738–1753, 2008.
- [21] H. Calisto, N. Martins, and N. Afgan, "Diagnostic system for boilers and furnaces using CFD and neural networks," *Expert Systems with Applications*, vol. 35, no. 4, pp. 1780–1787, 2008.
- [22] S. Lukić, Ž. Čojbašić, Z. Perić et al., "Artificial neural networks based early clinical prediction of mortality after spontaneous intracerebral hemorrhage," *Acta Neurologica Belgica*, vol. 112, no. 4, pp. 375–382, 2012.
- [23] D. Petković, M. Issa, N. D. Pavlović, L. Zentner, and Ž. Čojbašić, "Adaptive neuro fuzzy controller for adaptive compliant robotic gripper," *Expert Systems with Applications*, vol. 39, no. 18, pp. 13295–13304, 2012.
- [24] D. Petković, N. D. Pavlović, Ž. Čojbašić, and N. T. Pavlović, "Adaptive neuro fuzzy estimation of underactuated robotic gripper contact forces," *Expert Systems with Applications*, vol. 40, no. 1, pp. 281–286, 2013.
- [25] D. Petković, Ž. Čojbašić, and S. Lukić, "Adaptive neuro fuzzy selection of heart rate variability parameters affected by autonomic nervous system," *Expert Systems with Applications*, vol. 40, no. 11, pp. 4490–4495, 2013.
- [26] S. Timung and T. K. Mandal, "Prediction of flow pattern of gas-liquid flow through circular microchannel using probabilistic neural network," *Applied Soft Computing*, vol. 13, no. 4, pp. 1674–1685, 2013.
- [27] Z. Čojbašić, V. Nikolić, E. Petrović et al., "A real time neural network based finite element analysis of shell structure," *Facta Universitatis, Series: Mechanical Engineering*, vol. 12, no. 2, pp. 149–155, 2014.
- [28] S. Kar, S. Das, and P. K. Ghosh, "Applications of neuro fuzzy systems: a brief review and future outline," *Applied Soft Computing Journal*, vol. 15, pp. 243–259, 2014.
- [29] D. Brkić, "Can pipes be actually really that smooth?" *International Journal of Refrigeration*, vol. 35, no. 1, pp. 209–215, 2012.
- [30] N. Afzal, A. Seena, and A. Bushra, "Turbulent flow in a machine honed rough pipe for large Reynolds numbers: general roughness scaling laws," *Journal of Hydro-Environment Research*, vol. 7, no. 1, pp. 81–90, 2013.
- [31] D. H. Yoo and V. P. Singh, "Explicit design of commercial pipes with secondary losses," *Journal of Hydro-Environment Research*, vol. 4, no. 1, pp. 37–45, 2010.
- [32] D. Brkić, *Efficiency of distribution and use of natural gas in households (in Serbian) [Ph.D. dissertation]*, University of Belgrade, Belgrade, Serbia, 2010, <http://phaidrdbg.bg.ac.rs/o:7888>.
- [33] H. K. Winning and T. Coole, "Explicit friction factor accuracy and computational efficiency for turbulent flow in pipes," *Flow, Turbulence and Combustion*, vol. 90, no. 1, pp. 1–27, 2013.
- [34] D. Brkić, "New explicit correlations for turbulent flow friction factor," *Nuclear Engineering and Design*, vol. 241, no. 9, pp. 4055–4059, 2011.
- [35] X. Fang, Y. Xu, and Z. Zhou, "New correlations of single-phase friction factor for turbulent pipe flow and evaluation of existing single-phase friction factor correlations," *Nuclear Engineering and Design*, vol. 241, no. 3, pp. 897–902, 2011.
- [36] D. Brkić, "Discussion of "Gene expression programming analysis of implicit Colebrook–White equation in turbulent flow friction factor calculation" by Saeed Samadianfar [J. Pet. Sci. Eng. 92–93 (2012) 48–55]," *Journal of Petroleum Science and Engineering*, vol. 124, pp. 399–401, 2012.
- [37] Ž. Čojbašić and D. Brkić, "Very accurate explicit approximations for calculation of the Colebrook friction factor," *International Journal of Mechanical Sciences*, vol. 67, pp. 10–13, 2013.
- [38] O. Giustolisi, L. Berardi, and T. M. Walski, "Some explicit formulations of Colebrook–White friction factor considering accuracy vs. computational speed," *Journal of Hydroinformatics*, vol. 13, no. 3, pp. 401–418, 2011.
- [39] A. Kroll and H. Schulte, "Benchmark problems for nonlinear system identification and control using soft computing methods: need and overview," *Applied Soft Computing Journal*, vol. 25, pp. 496–513, 2014.
- [40] M. T. Hagan and M. B. Menhaj, "Training feedforward networks with the Marquardt algorithm," *IEEE Transactions on Neural Networks*, vol. 5, no. 6, pp. 989–993, 1994.
- [41] D. Brkić, "Comments on 'Experimental study of the pressure drop in the cathode side of air-forced open-cathode proton exchange membrane fuel cells' by Barreras et al., Int. J. Hydrogen Energy, 36 (13) 2011, 7612–7620," *International Journal of Hydrogen Energy*, vol. 37, no. 14, pp. 10963–10964, 2012.
- [42] T. K. Serghides, "Estimate friction factor accurately," *Chemical Engineering*, vol. 91, no. 5, pp. 63–64, 1984.
- [43] E. Romeo, C. Royo, and A. Monzón, "Improved explicit equations for estimation of the friction factor in rough and smooth pipes," *Chemical Engineering Journal*, vol. 86, no. 3, pp. 369–374, 2002.
- [44] A. R. Vatankhah and S. Kouchakzadeh, "Discussion of "turbulent flow friction factor calculation using a mathematically exact

- alternative to the colebrook–white equation” by Jagadeesh R. Sonnad and Chetan T. Goudar,” *Journal of Hydraulic Engineering*, vol. 134, no. 8, p. 1187, 2008.
- [45] D. Buzzelli, “Calculating friction in one step,” *Machine Design*, vol. 80, no. 12, pp. 54–55, 2008, http://images.machinedesign.com/images/archive/72728061908flui_00000051017.pdf.
- [46] D. J. Zigrang and N. D. Sylvester, “Explicit approximations to the solution of Colebrook’s friction factor equation,” *AIChE Journal*, vol. 28, no. 3, pp. 514–515, 1982.
- [47] L. K. Li, S. Shao, and K.-F. C. Yiu, “A new optimization algorithm for single hidden layer feedforward neural networks,” *Applied Soft Computing Journal*, vol. 13, no. 5, pp. 2857–2862, 2013.
- [48] J. D. Olden and D. A. Jackson, “Illuminating the ‘black box’: a randomization approach for understanding variable contributions in artificial neural networks,” *Ecological Modelling*, vol. 154, no. 1-2, pp. 135–150, 2002.
- [49] D. Brkić, “An explicit approximation of Colebrook’s equation for fluid flow friction factor,” *Petroleum Science and Technology*, vol. 29, no. 15, pp. 1596–1602, 2011.
- [50] A. R. Vatankehah, “Comment on ‘Gene expression programming analysis of implicit Colebrook–White equation in turbulent flow friction factor calculation,’” *Journal of Petroleum Science and Engineering*, vol. 124, pp. 402–405, 2014.
- [51] S. Samadianfard, “Gene expression programming analysis of implicit Colebrook–White equation in turbulent flow friction factor calculation,” *Journal of Petroleum Science and Engineering*, vol. 92-93, pp. 48–55, 2012.
- [52] S. Samadianfard, M. Taghi Sattari, O. Kisi, and H. Kazemi, “Determining flow friction factor in irrigation pipes using data mining and artificial intelligence approaches,” *Applied Artificial Intelligence*, vol. 28, no. 8, pp. 793–813, 2014.
- [53] B. J. McKeon, M. V. Zagarola, and A. J. Smits, “A new friction factor relationship for fully developed pipe flow,” *Journal of Fluid Mechanics*, vol. 538, pp. 429–443, 2005.
- [54] D. Brikić, “A gas distribution network hydraulic problem from practice,” *Petroleum Science and Technology*, vol. 29, no. 4, pp. 366–377, 2011.
- [55] Y. B. Dibike, D. Solomatine, and M. B. Abbott, “On the encapsulation of numerical-hydraulic models in artificial neural network,” *Journal of Hydraulic Research*, vol. 37, no. 2, pp. 147–161, 1999.
- [56] D. Brkić, “A note on explicit approximations to Colebrook’s friction factor in rough pipes under highly turbulent cases,” *International Journal of Heat and Mass Transfer*, vol. 93, pp. 513–515, 2016.
- [57] J. R. Sonnad and C. T. Goudar, “Turbulent flow friction factor calculation using a mathematically exact alternative to the Colebrook–White equation,” *Journal of Hydraulic Engineering*, vol. 132, no. 8, pp. 863–867, 2006.
- [58] B. J. Schorle, S. W. Churchill, and M. Shacham, “Comments on: ‘An explicit equation for friction factor in pipe,’” *Industrial & Engineering Chemistry Fundamentals*, vol. 19, no. 2, p. 228, 1980.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

