International Communications in Heat and Mass Transfer 76 (2016) 197–201

Contents lists available at ScienceDirect

International Communications in Heat and Mass Transfer

journal homepage: www.elsevier.com/locate/ichmt

Predicting bubble size and bubble rate data in water and in froth flotation-like slurry from computational fluid dynamics (CFD) by applying deep neural networks (DNN)☆

Gonzalo Montes-Atenas [a], Fabián Seguel [b], Alvaro Valencia [c,*], Sohail Masood Bhatti [d,1], Muhammad Salman Khan [d,2], Ismael Soto [b], Néstor Becerra Yoma [b]

[a] Minerals and Metals Characterisation and Separation (M^2CS) Research Group, Department of Mining Engineering, Universidad de Chile, Santiago, Chile
[b] Department of Electrical Engineering, Universidad de Santiago de Chile, Santiago, Chile
[c] Department of Mechanical Engineering, Universidad de Chile, Santiago, Chile
[d] Department of Electrical Engineering, Universidad de Chile, Santiago, Chile

ARTICLE INFO

Available online 26 May 2016

ABSTRACT

Accurate characterization of two phase bubbly flows is crucial in many industrial processes such as fluidized reactors, ore froth flotation, etc. The bubble size determines the rate at which components present in the gas phase are transferred to the surroundings and vice versa while bubble rate defines the appropriate bubbly flow regime occurring in the heterogeneous system. This research work employs deep neural networks (DNNs) to predict bubble size and bubble rate using data obtained from validated computational fluid dynamics (CFD) computations. Pure water and slurry (in conditions similar to those employed in mineral froth flotation) case studies are evaluated. It is found that the DNN can predict the CFD results accurately when using four hidden layers, describing discontinuities in the bubbly flow regime. The relative errors computed between the CFD data and the prediction obtained by the DNN is as low as 8.8% and 1.8% for bubble size and bubble rate, respectively. These results confirm that the DNN can be applied to sophisticated fluid dynamics systems and allow developing better control process strategies since once the DNN is trained critical variables can be computed very efficiently. The slurry case study, although restricted to the application of mineral froth flotation, can also be generalized to other industrial operations keeping the exact same procedure.

1. Introduction

Determining bubble size distribution and other characteristics of bubbly flows is a challenge and a necessity for a number of industrial operations such as mineral processing, oil industry, wine industry, wastewater treatment, fluidization reactors, bioreactors, etc. [1]. Particularly, in the field of mineral processing the search for new, improved and more accurate models to represent the efficiency of the froth flotation operation, one of the largest tonnage operations in the industry field, is still an ongoing process. Therein, it is widely accepted that the bubbly flow characteristics are key for a correct determination of the efficiency of the separation process and so far no analytical descriptions have been capable to gather all the information to understand the process. Although many models have been suggested as a possible representation of the froth flotation process and its sub-processes [2], they have proved to follow the trends the real systems exhibit. However, the uncertainty embedded in computing the efficiency of the operation is still an unresolved matter. Such uncertainty varies from 5 to 30% [3,4]. Computational fluid dynamics (CFD) tools are able to capture all complexities present in real froth flotation systems [5]. However, tracking the behavior of bubbly flows is an open problem.

The application of ordinary neural networks (NNs) has penetrated many applications fields dramatically and the mapping and prediction of fluid dynamic equations has been a highly fruitful field for scientific research [6]. One variation of NNs is the use of Deep Neural Networks (DNNs) which is one of the most recent developments in machine learning and represents a tremendous progress compared to the ordinary NN framework. DNNs are essentially neural networks with several (i.e. more than one) hidden layers that are pre-trained to reduce the limitations of the classic gradient based backpropagation training. Increasing the number of hidden layers improves the capability of the network to solve highly difficult, nonlinear and dynamic functions when compared to shallow networks. Nevertheless, to use a gradient-based

☆ Communicated by W.J. Minkowycz.
* Corresponding author.
[1] Current address: Department of Computer Science & IT, University of Lahore, Gujrat Campus, Pakistan.
[2] Current address: Department of Electrical Engineering, University of Engineering and Technology, Peshawar, Pakistan.

http://dx.doi.org/10.1016/j.icheatmasstransfer.2016.05.031
0735-1933/

Nomenclature

a	Main or major bubble axis, [m]
b	Secondary or minor bubble axis (perpendicular to "a"), [m]
d_b	Bubble diameter, [m^3]
d_0	Inlet diameter for gas in the system, [m]
k	Bias of the encoder of the DNN
$\widetilde{k}$	Bias of the decoder of the DNN
Q	Air flowrate, [m^3/s]
RMS	Root-mean-squared deviation
V_b	Bubble volume, [m^3]
X	Input vector of the DNN
$\hat{X}$	Estimation of X in the DNN pre-training
W	Weights of the encoder of the DNN in the pre-training
$\widetilde{W}$	Weights of the decoder of the DNN in the pre-training
μ_{slurry}	Dynamic viscosity of the slurry phase
μ_{fluid}	Dynamic viscosity of water
ρ_l	Water density, [kg/m^3]
ρ_{slurry}	Slurry density, [kg/m^3]
ρ_s	Solid density, [kg/m^3]
σ	Surface tension, [Nm]
φ_l	Water percent in the slurry phase
φ_s	Solids percent in the slurry phase
ϕ	Volume fraction of solids in the slurry phase

optimization strategy may be not effective when the gradient propagates across multiple nonlinearities. To avoid this limitation, in [7] it was presented a procedure which consists of pre-training one layer at a time, demonstrating that it is possible to learn in deep NN-architectures. In the recent years, the applications of this method with Restricted Boltzmann Machine have grown exponentially to address pattern recognition problems in computer vision [8], automatic speech recognition and natural language processing [9]. Surprisingly, despite the fact that DNNs have also been used for time series modeling and forecasting achieving promising results [10], the prediction problem has not been addressed exhaustively with DNNs.

2. Modeling CFD response with DNN

Reliable computational fluid dynamics (CFD) data are often used to simulate pressure and velocity fields in complex systems such as bubbly flows in different applications such as mineral froth flotation among others [11]. The CFD data and the uncertainties behind the use of turbulent frameworks collect features of realistic systems which may be validated with empirical and semi-empirical models [12,13]. In [14] experimental and CFD results of two-phase fluid flow in a tube were predicted using ordinary NN. Those results showed that the NN can predict with a reasonable accuracy the complex flow. In this study, the bubble size and bubble rate are obtained using a set of CFD data. The prediction considered employs the leave-one-out strategy where the whole dataset but one is used to train the DNN tool predicting the one not seen in the training step. This procedure is repeated by shifting the unseen data to the next one until completing the whole set of CFD data.

Fig. 1 shows the DNN architecture employed in this paper. In order to map both bubble size and bubble rate, two decoupled and independent DNNs are used. The first network is trained to learn the "hidden" function that estimates bubble size using the following inputs variables: initial velocity; fluid viscosity; fluid density; surface tension; and, contact angle. The second DNN is trained to learn the "hidden" function that outputs the bubble rate from the same set of inputs. It is worth highlighting that four out of the five input variables correspond to

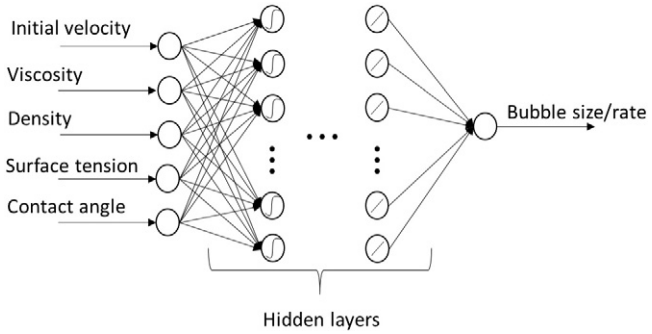

Fig. 1. The DNN architecture employed to learn the hidden function that delivers the bubble size and rate by making use of the following information: gas inlet velocity; fluid viscosity; fluid density; surface tension; and, contact angle.

fluid characteristics that are possible to be measured. The fifth input variable represents the inlet velocity computed with respect to the cross surface area of the nozzle throughout which the air injected into the simulated tank. The hidden layers activation functions are nonlinear (e.g. sigmoid), but the activation function of the output layer is linear. As a result of this procedure, the DNN architecture that is able to represent accurately the CFD results is reported.

3. Methodology

3.1. Details of CFD simulations

CFD simulations of bubbly flows were obtained from a 3D cylindrical system sketched in Fig. 2. It consists of a container 50 cm height and 20 cm diameter filled with fluid up to 40 cm. The gas flowrate enters at the bottom through an inlet located at the center of the base with a 2 mm diameter. A mesh of 600,122 cells was considered to resolve the mass balance and momentum equations according to the equations presented in a previous research work [15]. The fluid media considered in the research were pure water and slurry phase, where the latter is similar to that observed in mineral froth flotation operations, demonstrating the significant differences occurring when passing from a fluid free of solids and reagents (frother-free aqueous solution) to another with 30% solid with frother reducing its surface tension. The main properties of both fluids are presented as follows.

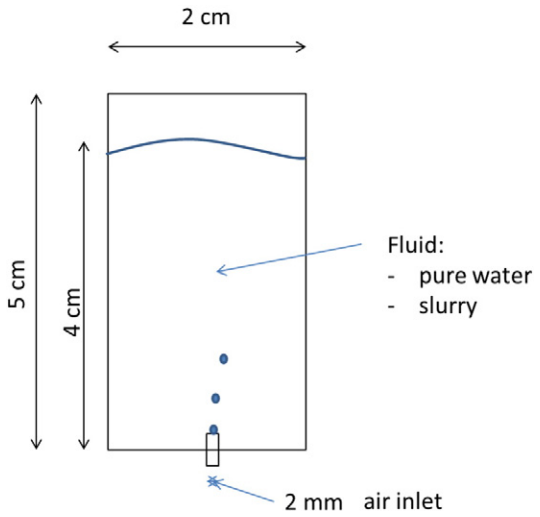

Fig. 2. Sketch of the CFD 3D cylindrical system used to obtain bubble volume and bubble rate.

3.1.1. Computation of slurry density from solid and liquid densities and their percentage in the mixture.

Eq. (1) is generally used to compute the slurry density.

$$\rho_{slurry} = \frac{100}{\frac{\varphi_s}{\rho_s} + \frac{\varphi_l}{\rho_l}} \tag{1}$$

where φ_s and φ_l represent the solids and liquid mass percents, and ρ_s and ρ_l are the densities of the solid and liquid pure phases. For example, in flotation tanks with roughing and scavenging duties the solids percent are about 40%. Assuming that the average ore density is 2700 kg/m^3, the slurry density is 1337 kg/m^3. The density used for pure water is 1000 kg/m^3.

3.1.2. Computation of slurry viscosity from solid and liquid densities, and their percentage in the mixture.

The viscosity equation for high volumetric concentrations of particles has been determined utilizing the equation [16]:

$$\mu_{slurry} = \mu_{fluid}\left(1 + 2.5\phi + 10.05\phi^2 + 0.00273e^{16.6\phi}\right) \tag{2}$$

where μ_{slurry} and μ_{fluid} represent the dynamic viscosity of the slurry phase and the dynamic viscosity of the pure liquid used, and ϕ is the volumetric fraction of solids present in the slurry phase.

3.1.3. Estimation of the surface tension and contact angle.

The surface tensions used for both liquids are summarized in Table 1. The surface tension of the slurry is lower than that of water with a value commonly reported in froth flotation experiments [17]. The contact angle of the aqueous phase with the vessel wall was considered constant and equal to 60° for both fluids.

3.2. Bubbly flow results

Fig. 3 presents the simulation results for an initial surface velocity of 0.175 [m/s] for pure water and slurry. As expected, bubbles are not perfect spheres. Nevertheless, when comparing both fluids at the same air flow rate the bubbles appear closer in the case of slurry phase, which can be explained by the strong difference in viscosity between the water and slurry.

3.3. Bubble volume estimation using image analysis

Volume of each bubble was estimated with two images captured from two different angles 90° apart. First, the boundary of the tank was extracted and the remaining image was removed. Next, the bubble edges were detected. The volume of fluid (VOF) is determined by the boundary of the bubble that is defined between 0 and 1. In this paper, the threshold to detect the bubble boundary was made equal to 0.5 [18]. After segmenting the bubbles from both images, volume was estimated. Based on the shape of the bubbles, an oblate spheroid model was fit by calculating the major and minor axes, a and b respectively, utilizing both images. The volume was then estimated according to (see Table 2):

$$V_b = \frac{4}{3}\pi ab^2 \tag{3}$$

Table 1
Surface tension values for water and slurry.

Fluid	Surface tension (Nm)
Water	0.0728
Slurry	0.0400

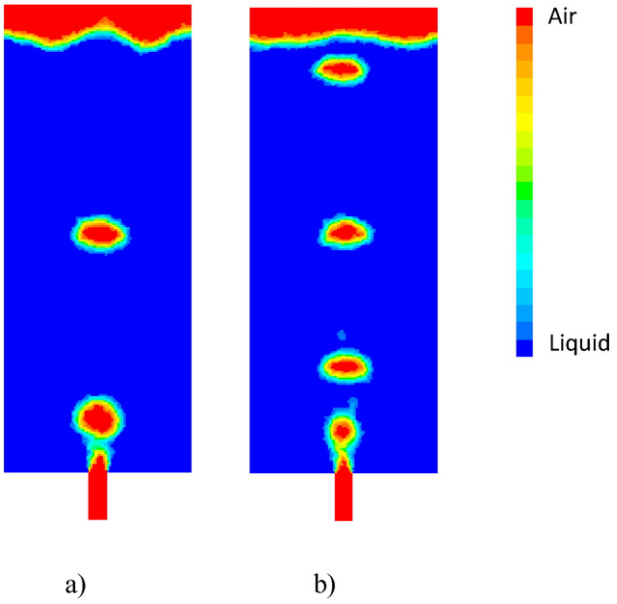

Fig. 3. Example of the bubbly flows in pure water and slurry phase for 0.175 [m/s] inlet gas velocity, a) water, b) slurry.

3.4. Validation of CFD results with experimental equations

Jamialahmadi and collaborators [19] summarized a number of equations for bubble size under different conditions. Table 3 indicates the empirical equations which explain the trends in bubble size obtained using CFD simulations and the fitting wellness. The equation reported by Gaddis and Vogepohl [20] describes correctly the linear trend observed in the case of pure water while the model presented by Van Krevelen and Hoftijzer [21] explain well the non-linear behavior observed in the case of the slurry phase. According to Jamialahmadi and collaborators [19] these equations have a relative error with respect to experimental data below 20%. Additionally, Table 3 indicates a good correlation between the CFD data and the equations validating these results. The average error in the case of slurry for bubble volume estimation was below 15%.

3.5. Deep neural network training

In order to train a "deep architecture" the greedy layer-wise approach was adopted to pre-train the DNN. Greedy layer-wise initializes the weights of the deep network by training each layer at a time. The first stage named encoder, takes an input vector X, $X \in \mathbb{R}^n \{0,1\}$ and maps the input into a hidden representation Y, $Y \in \mathbb{R}^q \{0,1\}$:

$$Y = f(WX + k) \tag{4}$$

Table 2
Average bubble volume and bubble rate obtained from CDF simulation.

Inlet gas velocity (m/s)	Average bubble volume (mm^3)		Average bubble rate (bubbles/s)	
	Water	Slurry	Water	Slurry
0.025	12.494	5.210	2.645	6.305
0.050	13.474	6.310	4.881	11.016
0.075	14.699	8.910	6.685	14.937
0.100	15.233	10.412	8.123	18.265
0.125	16.169	11.024	8.629	19.920
0.150	16.837	12.126	8.993	20.098
0.175	18.240	20.701	12.706	16.672
0.200	19.977	22.583	12.784	22.537
0.225	22.583	26.190	12.658	20.309
0.250	23.523	28.373	12.695	19.739

Table 3
Equations for bubble size used to validate the CFD results.

Fluid	Equation	Pearson coefficient (R^2)	Reference
Pure water	$d_b = \left[\left(\frac{6d_0\sigma}{\rho_{liquid}g}\right)^{4/3} + \left(\frac{81\nu Q}{\pi g}\right) + \left(\frac{135Q^2}{4\pi^2 g}\right)^{4/5}\right]^{1/4}$	0.994	[20]
Slurry phase	$d_b = \left(1.722\frac{Q^{5/6}\frac{6}{\pi}}{g^{3/5}}\right)^{1/3}$	0.962	[21]

where f is a nonlinear activation function (a sigmoid function in this particular case). Then, this representation is coded back with a decoder. The decoder function attempts to generate an estimation of the original feature input vector, $\hat{X}$, as expressed below:

$$\hat{X} = f\left(\widetilde{W}Y + \widetilde{k}\right) \tag{5}$$

The weights of the decoder, $\widetilde{W}$, are typically the transpose of the weights of the encoder, $\widetilde{W} = W^T$. The autoencoder (i.e. encoder + decoder) is trained by making use of the delta rule or back propagation method and by minimizing the cost function, i.e. the error between the inputs and its approximation, $l(X,\hat{X})$ is represented by Eq. (6).

$$minl\left(X,\hat{X}\right) = \left\|X - \hat{X}\right\|^2 \tag{6}$$

After pre-training the first layer, the procedure is repeated to all the hidden layers. Once the pre-training is complete, the training process of all the hidden layers takes place with the backpropagation algorithm.

4. Results and discussion

A DNN was trained with 19 sets of inputs–outputs; then, the prediction accuracy of the trained DNN was evaluated with the 20th set of input–output that was not seen in training. This scheme was repeated for each input–output set. There is no universal method to determine the optimal number of parameters for a network. It depends on the structure of the problem at hand and on the available training dataset. Different DNN architectures were tested in order to find the architecture with the lowest RMS error [22] for both bubble size and bubble rate.

The performances of DNNs with one, two, three and four hidden layers are compared. In addition, each one of the architectures tested was optimized with respect to the number of neurons on each hidden layer that was varied from four neurons to 20 neurons with a step equal to two. The pre-training and training epochs were set at 1000. The activation function in the hidden layers corresponds to a sigmoid, and the activation function in the output layer is linear. For the DNN with one hidden layer the optimum number of hidden neurons was equal to ten, with a RMS error of 0.0735 in bubble rate and 0.0549 in bubble size. The lowest RMS error of the DNN with two hidden layers was equal to 0.0715 for bubble rate (ten neurons in the first hidden layer and 12 neurons in the second hidden layer) and 0.0500 for bubble size (four neurons in the first hidden layer and 12 neurons in the second hidden layer). The lowest RMS error of the DNN with three hidden layers was equal to 0.0619 for bubble rate (ten neurons in the first and second hidden layers, and eight neurons in the third hidden layer) and 0.0324 for bubble size (12, eight and 12 neurons in the first, second and third hidden layers, respectively). The DNN architecture with the lowest RMS corresponded to the four hidden layer architecture. In this case, the RMS was equal to 0.0162 and 0.0534 for bubble size (12, six, eight and six neurons in the first, second, third and fourth hidden layers, respectively) and bubble rate, with the same number of hidden neurons, respectively. The relative error between the CFD data and the DNNs prediction is 8.8% and 1.8% for bubble size and bubble rate, respectively. Figs. 4 and 5 show the original CFD results and the predicted ones by the four hidden layer DNNs for bubble size and bubble rate, respectively.

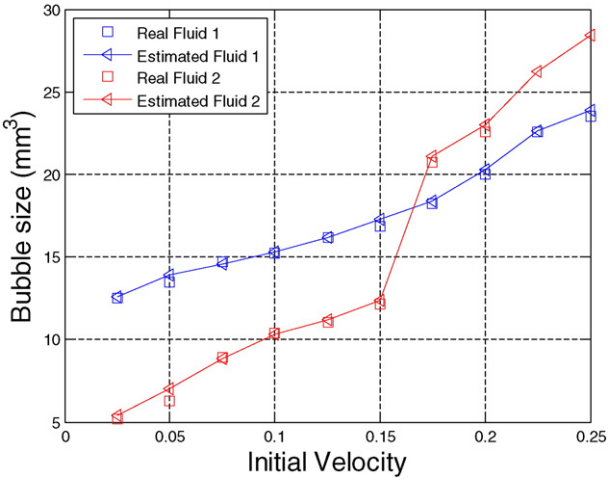

Fig. 4. Four hidden layer DNN performance for bubble size estimation.

The computer used to train and test the DNNs had an Intel i5 1.5 GHz microprocessor with 4 GB RAM. The training of the four hidden layer DNNs, including all the configurations of hidden neurons, required 25 h of computation. However, the testing time was just 0.18 s ensuring a fast evaluation of the bubbling operation which confirms its high potential usefulness for designing improved control strategies of industrial processes having bubbly flows in their implementation in fields such as, but not restricted to, mineral processing with an adequate uncertainty.

5. Conclusions

The DNN framework predicted accurately the bubble size and bubble rate obtained from CFD simulations. The relative errors between CFD data and DNN predictions for bubble size and bubble rate are as low as 8.8% and 1.8%, respectively, indicating that the technique is suitable to estimate these features of bubbly flows system. Moreover, the four hidden layer DNN allows describing the whole range of bubble sizes and bubble rates tested including discontinuities in the bubbly flow regime. These results corroborate that the DNN can be applied to sophisticated fluid dynamic problems such as slurries in conditions close to those employed in the mining industry in froth flotation operations. This procedure can also be generalized to other industrial

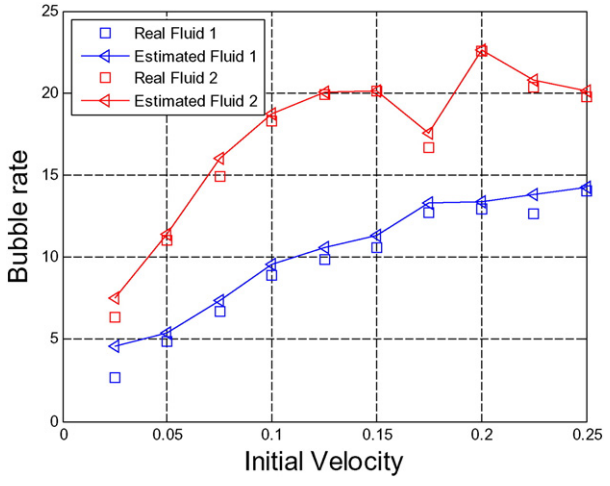

Fig. 5. Four hidden layer DNN performance for bubble rate estimation.

operations enabling the development of better process control strategies because once the DNN is trained critical variables can be computed very efficiently.

Acknowledgments

The research reported here was funded by CONICYT-Chile, project PIA ACT 1120.

References

[1] R.H. Perry, D.W. Green, Perry's Chemical Engineers' Handbook, seventh ed. McGraw Hill, Salt Lake city, Utah, 1997 14.69–14.79.
[2] R.P. King, Modeling and Simulation of Mineral Processing Systems, second ed. Society for Mining, Metallurgy, and Exploration, Inc., Englewood Co, 2012 341–413.
[3] F.F. Pitard, Pierre Gy's Sampling Theory and Sampling Practice. Heterogeneity, Sampling Correctness and Statistical Process Control, second ed. CRC Press, Boca Raton, London, New York, Washington D.C., 1993 315–328.
[4] G. Sandoval-Zambrano, G. Montes-Atenas, Errors in the estimation of size-by-liberation flotation rate constants, Miner. Eng. 27-28 (2012) 1–10.
[5] T.J. Chung, Computational Fluid Dynamics, first ed. Cambridge University Press, Cambridge, New York, Melbourne, Madrid, Cape Town, 2002 902–924.
[6] M. Rahimi, M. Hajialyani, R. Beigzadeh, et al., Application of artificial neural network and genetic algorithm approaches for prediction of flow characteristic in serpentine microchannels, Chem. Eng. Res. Des. 98 (2015) 147–156.
[7] G.E. Hinton, S. Osindero, Y.-W. Teh, A fast learning algorithm for deep belief nets, Neural Comput. 18 (7) (2006) 1527–1554.
[8] A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural networks, Adv. Neural Inf. Proces. Syst. 25 (2012) 1097–1105.
[9] G. Hinton, L. Deng, D. Yu, G. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. Sainath, B. Kingsbury, Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups, IEEE Signal Process. Mag. 29 (6) (2012) 82–97.
[10] M. Längkvist, L. Karlsson, A. Loutfi, A review of unsupervised feature learning and deep learning for time-series modeling, Pattern Recogn. Lett. 42 (2014) 11–24.
[11] P.T.L. Koh, M.P. Schwarz, CFD modelling of bubble-particle attachment in flotation cells, Miner. Eng. 19 (2006) 619–626.
[12] A. Valencia, M. Cordova, J. Ortega, Numerical simulation of gas bubbles formation at a submerged orifice in a liquid, Int. Commun. Heat Mass Transf. 29 (6) (2002) 821–830.
[13] I. Chakraborty, G. Biswas, S. Polepalle, P.S. Ghoshdastidar, Bubble formation and dynamics in a quiescent high density liquid, AIChE J. 61 (11) (2015) 3996–4012.
[14] A. Alizadehdakhel, M. Rahimi, J. Sanjari, A.A. Alsairafi, CFD and artificial neural network modeling of two-phase flow pressure drop, Int. Commun. Heat Mass Transf. 36 (2009) 850–856.
[15] M.S. Khan, G. Montes, A. Valencia, S.M. Bhatti, N. Becerra Yoma, On discriminating sizes of CFD generated bubbles with signal processing analysis, Int. J. Heat Mass Transf. 89 (2015) 996–1006.
[16] D.G. Thomas, Transport characteristics of suspensions. VIII: a note on the viscosity of Newtonian suspensions of uniform spherical particles, J. Colloid Sci. 20 (1965) 267–277.
[17] R.T. Rodrigues, J. Rubio, New basis for measuring the size distribution of bubbles, Miner. Eng. 16 (2003) 757–765.
[18] J. Klostermann, K. Schaake, R. Schwarze, Numerical simulation of a single rising bubble by VOF with surface compression, Int. J. Numer. Methods Fluids 71 (8) (2013) 960–982.
[19] M. Jamialahmadi, M.R. Zehtaban, H. Muller-Steinhagen, A. Sarrafi, J.M. Smith, Study of bubble formation under constant flow conditions, Trans. IChemE 79 (A) (2001) 523–532.
[20] E.S. Gaddis, A. Vogelpohl, Bubble formation in quiescent liquids under constant flow conditions, Chem. Eng. Sci. 41 (1986) 97–105.
[21] D.W. van Krevelen, P.J. Hoftijer, Studies of gas bubble formation-calculation of interfacial area in bubble contactors, Chem. Eng. Prog. 46 (1950) 29–35.
[22] G. Acuña, E. Pinto, Development of a Matlab Toolbox for the design of grey-box neural models, Int. J. Comput. Commun. Control 1 (2) (2006) 7–14.