Recognition of an obstacle in a flow using artificial neural networks

Mauricio Carrillo, Ulices Que,[*] José A. González, and Carlos López

Laboratorio de Inteligencia Artificial y Supercómputo, Instituto de Física y Matemáticas, Universidad Michoacana
de San Nicolás de Hidalgo, Edificio C-3, Cd. Universitaria, 58040 Morelia, Michoacán, México

In this work a series of artificial neural networks (ANNs) has been developed with the capacity to estimate the size and location of an obstacle obstructing the flow in a pipe. The ANNs learn the size and location of the obstacle by reading the profiles of the dynamic pressure q or the x component of the velocity v_x of the fluid at a certain distance from the obstacle. Data to train the ANN were generated using numerical simulations with a two-dimensional lattice Boltzmann code. We analyzed various cases varying both the diameter and the position of the obstacle on the y axis, obtaining good estimations using the R^2 coefficient for the cases under study. Although the ANN showed problems with the classification of very small obstacles, the general results show a very good capacity for prediction.

I. INTRODUCTION

Obstructions in tubes transporting fluids are a problem of high impact in modern society, as they are present in people's daily lives as well as in urban pipe networks [1], public health facilities[2], and industrial engineering [3]. On one hand, the accelerated urban and industrial growth in modern cities implies that obstructions in pipe networks are a very common problem, requiring a prompt reaction to solve it. These blockages can be caused by chemical or physical residues, as well as by structural defects of different kinds. Pipes are one of the most common ways of transporting fluids in the energy [4], chemical [5], manufacturing [6,7], and water [8] industries, as well as in houses, buildings, and sewage systems [9–11]. Clearly, in all these cases, the requirements of obstructed fluid delivery are crucial.

On the other hand, in the health-care sector, medical problems due to obstructions and/or blockages in the innumerable conduits that transport biological flows around the human body are very frequent and, in many cases, can be fatal. In biological conduits, blockages can be caused by previous surgeries, foreign bodies, infections, and deformations, among many other factors. Thus, cases such as obstructions in the digestive system [12,13] or in the cardiovascular system [14,15] are related to medical emergencies, which in many cases may involve the use of invasive clinical procedures, and therefore they require prompt attention.

All these scenarios motivate the scientific interest in the detection and comprehension of the shape and location of objects blocking or obstructing flows [16]. In this work, we use artificial neural networks (ANNs) as a flow pattern categorization in order to recognize these obstructions to fluid flow in a two-dimensional (2D) conduit. In this context, research studies such as [17–19] have proposed different methodologies to identify obstacles, leaks, or defects inside industrial or urban pipes. In particular, ANNs as machine learning methods have been applied to problems of fluid dynamics mainly in flow phase pattern identification, for example, in [20] and [21]. They have also been used as a tool for faster computational fluid

simulations and turbulence prediction [22–24] or for defect classification in tubular structures using images [25].

With this motivation, the objective of this work is to recognize the shape and location of an obstacle obstructing a pipe, whose dimensions were chosen considering pipes and networks that transport fluids for use in industrial and urban systems. For this, we have trained ANNs using physical information from the flow as input data. To that end, we have developed a generic 2D lattice Boltzmann numerical code [26,27], to simulate the flow of a fluid around an obstacle, contrasting the numerical solution with the benchmark [28]. Our problem takes into account different scenarios: changing the diameter and location of the obstacle, viscosity, and initial flow velocity and obtaining relevant physical information such as the velocity, vorticity, and dynamic pressure of the fluid along the numerical domain. In particular, we have considered the x component of the velocity field of the fluid (v_x) or the dynamic pressure (q) as the fundamental information to be analyzed by the ANN. We chose a ratio between the width of the pipe and the width of the immersed obstacle from $1/80$ to almost a value of 1; this is relevant in physical scenarios where the flow is disrupted by obstructions that can range from small blockages up to complete blockage of the pipe.

The content of the article is as follows: in Sec. II we present a detailed description of the problem of blockage in simulations with the lattice Boltzmann method (LBM). In Sec. III we explain the methodology followed in the cases under study, describing the results obtained in Sec. IV. Finally, in Sec. V we present our conclusions and directions for future work.

II. NUMERICAL SIMULATIONS

The simulation of 2D flow around an obstacle was performed by constructing a numerical code based on the lattice Boltzmann method as in Ref. [29]. The LBM is very popular because it is easy to implement and it has a high capacity to perform computational simulations in a wide variety of physical problems [30–32], mainly applied in computational fluid dynamics [33,34].

For our cases under study, we consider a cylindrical obstacle immersed in an infinite medium (free flow), with flow moving in the positive x direction. The cylindrical obstacle is
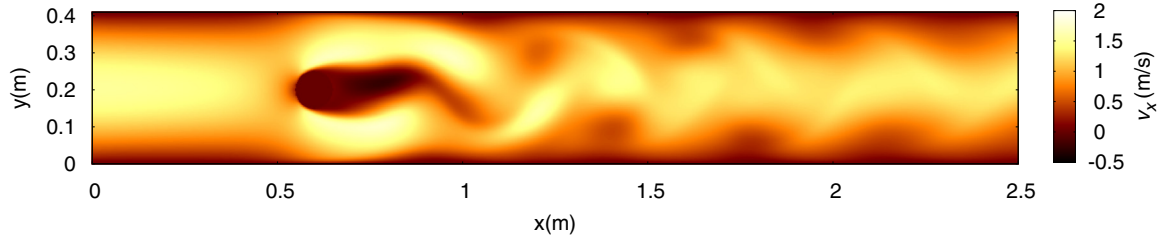
---

*qsalinas@ifm.umich.mx

FIG. 1. Magnitude of v_x for the flow around a cylindrical obstacle with an incoming Poisseuille flow with $v_c = 1.5$ m/s. The obstacle is fixed at $x = 0.6$ m and the fluid flows over the x-positive direction along the pipe. Note that the vortices form after the obstacle is driven along the direction of the flow, generating the characteristic Karman vortex street.

perpendicular to the x-y plane and, therefore, is represented by a circle in the 2D simulation. We impose boundary conditions at the outlet of the flow far enough such that the characteristic flow parameters are not affected by the internal calculations [35]. For the solid walls of the pipe and obstacle, we employed a full bounce-back boundary condition [26]. The boundary condition representing the incoming fluid was set up with a given velocity profile as input for the numerical modeling, such that the flow after the obstacle presents a pattern related to the input velocity profile, the diameter of the obstacle, and its location.

We define β as the value of the ratio of the diameter of the cylindrical obstacle to the width of the pipeline. The diameter of the obstacle is changed for values of β ranging from $\beta = 0.0122$, representing small blocking elements, up to values close to $\beta = 1$, representing obstructions of almost the total size of the diameter of the pipe. For the domain of the simulation, a mesh of 165×1000 nodes was used. Although the LBM code has dimensionless units, the system has been adapted to physical dimensions following Ref. [28]. We have chosen the physical units such that the numerical domain corresponds to a total length of $L_y = 0.41$ m in the vertical direction and $L_x = 2.5$ m along the horizontal. We considered a Poiseuille incoming fluid flow in a stationary regime with a density of $\rho = 10^3$ kg/m^3 and a kinematic viscosity of $\nu = 10^{-3}$ m^2/s, as used in Ref. [29]. The location of all the studied obstacles is at $x = 0.6$ m of the pipe, and their position on the y axis is described in the following section.

We performed a large number of different numerical simulations, considering as free input parameters the inlet velocity profile, the diameter of the obstacle (changing the values of β), and the position of the obstacle with respect to

the y axis. An example of a numerical simulation is shown in Fig. 1, with an obstacle of size $\beta = 0.244$ and a Poiseuille flow with a characteristic velocity of $v_c = 1.5$ m/s. The numerical simulations were stopped until the system reached a neutral stability, which occurred before the completion of 30 000 iterations, or in a physical time of approximately 16 s.

III. METHODOLOGY

In this work, we estimate the size and location of an obstacle by measuring v_x and q after the obstacle. On this matter, one could propose the simplest case: considering a single sensor and applying a linear regression between the obstacle diameter and the flow velocity at the location of the sensor. Although the linear regression is as good as the ANN in this case, the analyses in other scenarios show that the ANN outperforms the linear regression, and that is why we present only the ANN results. Moreover, the intention is to provide the first step of a methodology capable of estimating multiple and more complex obstacles or morphologies with nonsymmetrical shapes. For this, we have defined a *target region* around the area where the obstacle is immersed. In this region the size and position of the obstacle are estimated in terms of the proportion of the obstacle and fluid surrounding it. We have also set different numerical sensors across the pipe at distinct measuring sites along the x axis; a schematic of this is shown in Fig. 2. The measurement sites are located at $x = 1.75, 2.10,$ and 2.45 m, which we refer to as A, B, and C, respectively.

With this in mind, the ANNs were selected because they are flexible in terms of input data and they are easy to implement, outperforming linear regressions. In possible future work, we
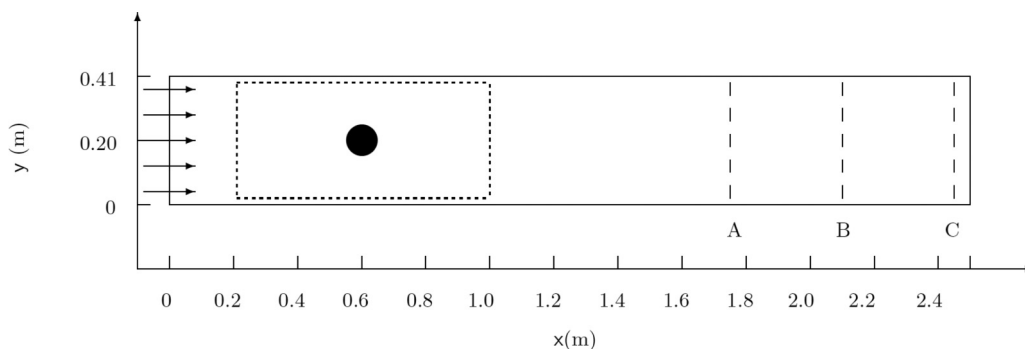

FIG. 2. Schematic of the 2D pipe, the cylindrical obstacle, and different measuring sites across the pipe, located at A = 1.75 m, B = 2.10 m, and C = 2.45 m, shown by dashed lines. The region within the dotted rectangle is the area used as target for the ANN.

would like to increase the complexity of the problem, trying other, more sophisticated machine learning methods.

In the next subsections we describe the cases under study, the structure of the ANN, and the methodology used for the input and target data, as well as the selection of the training and validation sets, which are used to adjust the ANN parameters, and the prediction set, for which the results are shown.

A. Database constructions and cases under study

The ability of the ANNs to predict the size and location of the obstacles is tested in three major cases:

(1) We perform 80 different numerical simulations for a 2D cylindrical obstacle immersed in the flow, located in the center of the y axis, i.e., at half the pipe diameter. We simulate obstacles with diameters ranging from $d = 0.005$ m to $d = 0.395$ m in steps of 0.005 m, i.e., values of β ranging from $\beta = 0.012$ to $\beta = 0.964$ in steps of 0.0122, plus a tiny obstacle of 0.001 m. From these cases, we select as the prediction set those obstacles with diameters ranging from $d = 0.02$ m to $d = 0.395$ m at intervals of $\Delta d = 0.025$ m. Meanwhile, the validation set consists of the obstacles from $d = 0.025$ m to $d = 0.375$ m, with $\Delta d = 0.025$ m also. The rest of the 49 obstacles are used as the training set. This case is divided into three subcases:

(a) The profile of v_x or q at $t = 16$ s, at the end of the numerical evolution, is considered the input vector. For simplicity in terms of the ANN structure, the information was extracted from only 83 of the 165 nodes of the numerical mesh. This approach is examined to study whether extracting the physical data, such as v_x or q, available at a fixed time and distance is enough to give a proper estimation of the obstacle's size.

(b) The time evolution of v_x or q over 300 time steps, from $t = 0$ to $t = 16$ s, measured at a single sensor at the center of the pipe, i.e., $y = 0.210$ m, is considered the input for the ANN, using the symmetry of the conduit. With this, we inspect the limits of the predictions considering the smallest number of sensors possible, with the advantage that several measurements are made over a lapse of time.

TABLE I. Parameters used in simulations for the cases studied. The column "Obstacles" lists the number of different diameters of the obstacles. The column "y positions" lists the number of different locations of the center of the obstacles along the y axis. The column "Flow velocities" refers to the number of different incoming flow velocities used in scenario 3. Meanwhile, the column "Space" is related to the number of equidistant sensors implemented along a measurement site. Finally, the column "Time" lists the number of time steps extracted in the numerical evolution performed on each considered numerical sensor.

Case	Obstacles	y positions	Flow velocities	Space	Time
1a	80	1	1	83	1
1b	80	1	1	1	300
1c	80	1	1	3	300
2a	4	43	1	83	1
2b	4	43	1	1	300
2c	4	43	1	3	300
3	11	1	12	10	1

TABLE II. Database for diameters, incoming flow velocities, and sensor locations used in case 3. The last column lists the locations of the 10 sensors on the y axis on the LBM mesh; in parentheses are listed their equivalent values in physical units. The values for v_x and q were extracted from these numerical sensors at both $x = 0$ m and $x = 2.10$ m on the pipe; the latter corresponds to measurement site B.

β	Velocity (m/s)	Sensor location
0.0122	0.15	11 (0.025 m)
0.0976	0.30	27 (0.065 m)
0.1952	0.45	43 (0.105 m)
0.2928	0.60	59 (0.145 m)
0.3904	0.75	75 (0.185 m)
0.4880	0.90	91 (0.225 m)
0.5856	1.05	107 (0.265 m)
0.6832	1.20	123 (0.305 m)
0.7808	1.35	147 (0.365 m)
0.8784	1.5	163 (0.405 m)
0.9638	1.65	
	1.8	

(c) This is the same procedure as case 1, but in addition, we add two equidistant sensors, i.e., we have three sensors, located at $y = 0.105$, 0.210, and 0.315 m. In this case, we study whether increasing the number of sensors, compared with the previous one, increases the prediction's accuracy.

(2) In contrast with case 1, we consider only three obstacle sizes, with values of $\beta = 0.122$, 0.244, and 0.488, that is, $d = 0.05$, 0.1, and 0.2 m. In every simulation, we change the position of the obstacle among 43 positions on the y axis, such that the obstacle can be either near the center of the pipe or near its walls, providing a total of 139 simulations. For each obstacle size we select 22 simulations for the training and validation sets, and the remaining 21 simulations, which are equally spaced on the y axis, are used for the prediction set. In order to prove the capacity of the ANNs for the estimation of a completely unknown obstruction size we include in the prediction set 21 simulations equally spaced on the y axis for with an obstacle diameter of $\beta = 0.366$. This scenario was also divided into three subcases following the same descriptions as in cases 1a, 1b, and 1c.

(3) In this case, we analyze a situation similar to case 1 where the obstacle is again located at $y = 0.210$ m and $x = 0.6$ m, with the difference that we change the characteristic velocity of the incoming flow from $v_c = 1.5$ m/s, used in cases 1 and 2, to different values ranging from $v_c = 0.15$ m/s to $v_c = 1.8$ m/s, in steps of $\Delta v_c = 0.15$ m/s. This analysis is performed to examine an extension of case 1, for different input flow velocities and 11 obstacle sizes as listed in Table II. In addition, we also explore the behavior of the network by adding more information about the incoming fluid and fewer sensors than in case 1. For simplicity, we consider 10 equidistant values on the y axis of v_x or q before the obstacle, at $x = 0$ according to the schematics in Fig. 2, and another 10 equidistant values of v_x or q at measurement site B, analogously as in case 1. With these arguments, the input data for the ANNs consist of 20 values for each of the 132 simulations produced. In order

to explore the performance of the network and its dependence on the choice of the sample sets for training, validation, and prediction, we now select the patterns randomly. The validation and prediction sets have 20 patterns each, while the training set has 92.

All the study cases simulated are summarized in Tables I and II.

B. Target region

As we use ANNs trained with supervised learning, we need to provide targets related to the true obstacle's size, shape, and location. For this, we define a region inside the pipe containing the obstacle and its immediate surroundings as the target region. However, if we consider each one of the nodes in the LBM simulation in the target region to be the objective for the ANN, as illustrated in Fig. 2, it will have a huge number of outputs, making it computationally expensive. Therefore, as the first approach we propose considering a target region consisting of a fixed mesh of 40×20 cells, where each cell consists of 64 nodes of the numerical mesh. Henceforth we refer to the target region as the *target grid*.

In addition, we assign a numerical value to each cell, depending on the relation of nodes that represent fluid or solid elements in the numerical mesh. This means that the proportion of target cells occupied by the obstacle is represented by the number of solid nodes divided by the total number of nodes in the numerical mesh contained in that cell. In other words, we have defined an occupation index for each cell in the target grid, where the solid-liquid ratio in the cell was calculated, meaning that a value of 1 represents a cell containing only solid elements of the obstacle and a value of 0 implies that there is only fluid in the cell. From now on, we call this index the solid-liquid ratio index (SLRI). An example of the transformation from the LBM simulation of the numerical mesh to the target grid for an obstacle of size 0.2 m is presented in Fig. 3, where the dark tones refer to cells mostly occupied by the obstacle (high SLRI), whereas clearer tones indicate cells containing a greater proportion of fluid elements (low SLRI).

C. Neural network structure

Let us recall that for case 1a, only half of the spatial nodes in the y direction of the lattice are selected as inputs for the ANN, reducing the number of points at which the vector fields are measured from 165 to 83, simplifying the input data and the structure of the ANN, and speeding up the computations. Therefore the input vector consists of 83 neurons related to the profile of the fluid at the corresponding measurement site. For example, if we consider the v_x values at the sensors at any measurement site, the input pattern is

$$I = \{v_{x_1}, v_{x_3}, \ldots, v_{x_i}, \ldots, v_{x_{165}}\}, \tag{1}$$

where $i = 1, 3, \ldots, 165$ indexes the down-sampling from 165 to 83 for nodes in the LBM simulation mesh. For case 1, the input vector is defined by the time series of v_x or q at $y = 0.21$ m during 300 time steps represented by the index (t); for example, for v_x this is

$$I = \{v_{x_{85}, t_1}, v_{x_{85}, t_2}, \ldots, v_{x_{85}, t_{300}}\}, \tag{2}$$
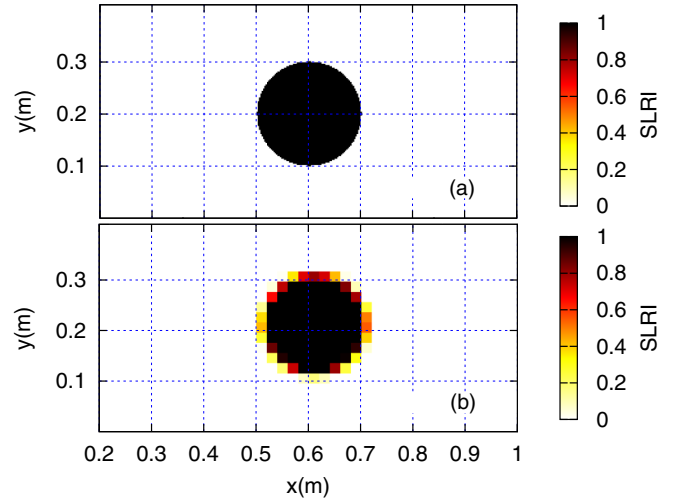

FIG. 3. (a) Plot representing the solid and fluid elements in the LBM mesh simulation; (b) plot representing the transformation from the simulation to the target grid. The lower number of cells used in the target grid in comparison with the LBM mesh causes a loss of resolution in the grid. The color box represents the solid-liquid ratio index (SLRI), where a 0 value means that a cell is composed of only fluid elements and 1 represents a cell fully occupied by solid elements. The grid lines shown are for reference only and do not represent the size of a target cell.

where $v_{x_{85}, t}$ labels the value of v_x at node 85 of the simulation where $y = 0.210$ m, i.e., at the middle of the pipe at a certain time t. For case 1c, the inputs of the ANN consist of the values of 300 time steps for v_x or q at the positions $y = 0.105, 0.210$, and 0.315 m, at the considered measurement site. In Fig. 4 we present the time series evolution for v_x at measurement site B. The inputs corresponding to this scenario are

$$I = \{v_{x_{43}, t_1}, v_{x_{85}, t_1}, v_{x_{127}, t_1}, \ldots, v_{x_{43}, t_{300}}, v_{x_{85}, t_{300}}, v_{x_{127}, t_{300}}\}. \tag{3}$$

In the second scenario, the inputs are the same as described in Eqs. (1)–(3). In case 3, let us recall that 10 values at each
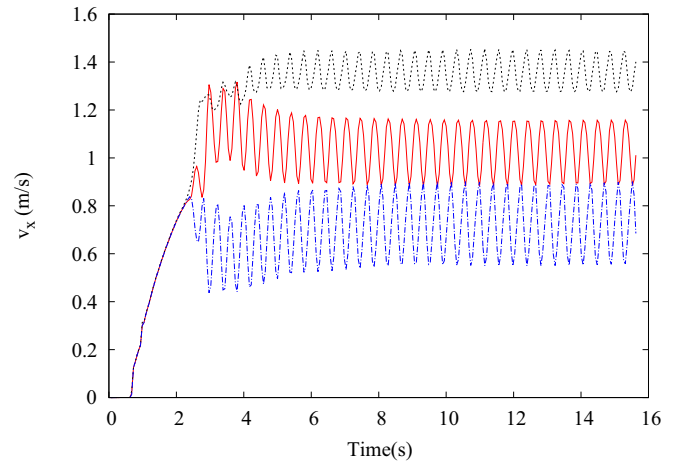

FIG. 4. Time series of the x component of the flow velocity at $y = 0.105$ m (solid curve), 0.210 m (dashed curve), and 0.315 m (dash-dotted curve) at measurement site B, for an obstacle of $\beta = 0.488$ centered at $y = 0.21$ m.

of $x = 0$ and $x = 2.1$ m (measurement site B) are considered as the input pattern. For example, following Table II, an input pattern using v_x is defined as

$$I = \{v_{x_{11}}(x = 0), v_{x_{27}}(x = 0), \ldots, v_{x_{163}}(x = 0), \ldots,$$
$$v_{x_{11}}(x = 2.1 \text{ m}), \ldots, v_{x_{163}}(x = 2.1 \text{ m})\}. \quad (4)$$

The same set of Eqs. (1)–(4) is applied when the dynamic pressure q is used instead of v_x as input data for the network.

The internal structure of the ANN can differ internally in each case, i.e., the number of inputs and hidden neurons, but the number of outputs is constant for all the cases, since the objective is the same: approximate the shape of an obstacle blocking flow. Recall that the region where the obstacle is located is described by 40×20 cells, which means that the target and the prediction have 800 elements. In a vectorized form, for an input pattern I, the result computed by the ANN is $\vec{O} = [O_1, O_2, \ldots, O_{800}]$, and the kth element of this vector is calculated by

$$O_k = F_2\left(\sum_{j=1}^{J} \sigma_{jk} F_1\left(\sum_{i=1}^{I} \tilde{\sigma}_{ij} \tilde{I}_i + \tilde{\sigma}_{0j}\right) + \sigma_{0k}\right), \quad (5)$$

where $1 \leqslant k \leqslant 800$ refers to the cells of the target grid, $\tilde{I}_i$ is the ith element of the input vector, and J is the number of hidden neurons. F_1 and F_2 are the activation functions for the hidden and output layers, respectively; $\tilde{\sigma}_{jk}$ and $\tilde{\sigma}_{0j}$ are the weights and bias terms between the input and the hidden layer; and σ_{jk} and σ_{0k} are the weights and bias terms between the hidden and the output layers.

The numerical implementation of the ANNs was developed from scratch using Fortran 90. Instead of using open source codes, we decided to use our own implementation to have full control of the details in the code, searching different structures and parameters in the learning process. On one hand, the selection of the ANN structure was done considering that it should be kept as simple as possible, in order to maintain the computational advantage, and complex enough for its adaptation to unknown patterns. In our cases we found that ANNs with an input layer as defined in Eqs. (1)–(4), i.e., one hidden layer with 20 neurons and an output layer with 800 neurons, were complex enough to give good results without loss of performance in all the cases under study. All the ANNs used have hidden and output layers with sigmoid activation functions. On the other hand, the ANNs were trained using a *backpropagation* algorithm [36]. In this work, we use this method to minimize the mean square error function, using a learning rate of 0.001, with a maximum of 15 000 iterations in training, and using a cross-validation technique as the stopping criteria. For clarity, all the results reported here correspond to the prediction set. For more details about supervised training and the backpropagation algorithm the reader can consult [37] and [38].

IV. RESULTS

In order to estimate the ANN prediction accuracy in each of the cases analyzed in the test set, we employ the R^2 coefficient. The calculation of R^2 was performed over the target and predicted grids, considering the real and predicted SLRI. R^2
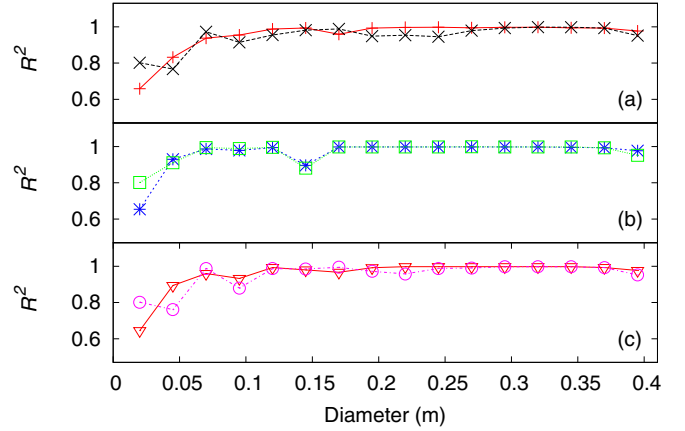

FIG. 5. R^2 coefficient for predictions produced by the ANN for obstacles located at the center of the pipe with different diameters. The ANN is trained with data extracted with a sensor located over the line in B and tested with measurements at A, B, and C. (a) The red curve with crosses shows the results using v_x, and q is represented by the black curve with X's at site A. (b) Input data were extracted at measurement site B, and the blue curve with asterisks and the green curve with squares represent R^2 using the profiles v_x and q, respectively. (c) The red curve with triangles and the magenta curve with circles correspond to the results for v_x and q at measurement site C, respectively. All R^2 coefficients are above 0.6, which can be interpreted as a good correlation between the prediction and the target. Note how the results are independent of the measurement location.

is defined as

$$R^2 = 1 - \frac{\sum_{i=1}(O_i - \langle T \rangle)^2}{\sum_{i=1}^{800}(T_i - \langle T \rangle)^2}, \quad (6)$$

where T_i and O_i are the ith target and the ANN output, respectively, and $\langle T \rangle$ is the average of the SLRI for the target vectors:

$$\langle T \rangle = \frac{1}{800} \sum_{i=1}^{800} T_i. \quad (7)$$

This means that the R^2 coefficient range is $(-\infty, 1]$, where a value of $R^2 = 1$ implies a perfect match term by term, between the target and the ANN prediction, while $R^2 \to 0$ means that the prediction approaches $\langle T \rangle$.

Let us recall that the measurements of the fluid flow consist of one snapshot of the profile of the v_x or q of the fluid, at the time when the system reaches neutral stability. For case 1, we present the results obtained from measurements by detectors located not only at the training site B, but also at sites A and C as shown in Fig. 5. In this case, we study how the ANN behaves if it is only trained with information at measurement site B and tested at sites A, B and C, for the same obstacles.

For measurements made at site B, both q and v_x show R^2 values very close to 1.0 for obstacles with diameters greater than 0.05 m, that is, for $\beta > 0.25$; see Fig. 6 for an example, where the target and predicted grids are plotted for an obstacle of $\beta = 0.4758$, achieving a prediction of $R^2 = 0.979$. However, for small obstacles the accuracy decreases, for example, with $\beta = 0.0488$ we obtained $R^2 = 0.654$ considering v_x and $R^2 = 0.802$ using q.
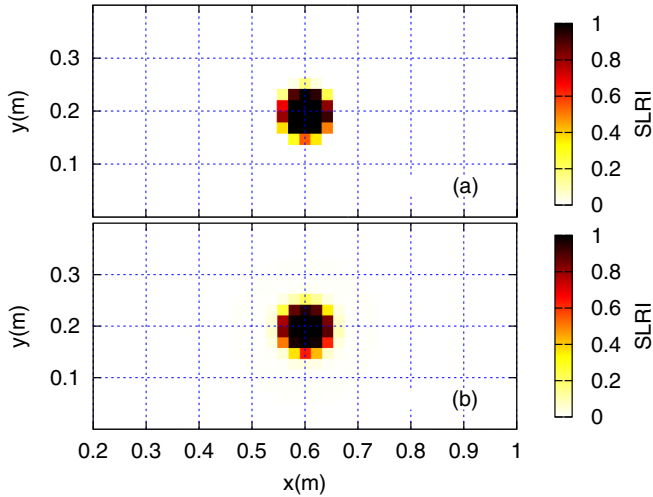
FIG. 6. (a) Target and (b) ANN prediction considering the profile of the v_x of the fluid generated by an obstacle at the center of the pipe with $\beta = 0.4758$. The color box represents the SLRI. The difference between the target and the prediction obstacle is almost imperceptible to the naked eye; the R^2 coefficient has reached a value of 0.979.

FIG. 7. ANN prediction analysis by R^2 for several obstacle sizes and centered on the pipe. Measuring the time series on a single sensor for v_x (crosses) and q (X's) at $y = 0.205$ m and three detectors, at $y = 0.105$, 0.210, and 0.315 m, for v_x (asterisks) and q (squares) at measurement site B. Note that by considering the time series of the three detectors the values of R^2 are close to 1 for diameters greater than $\beta = 0.20$, while considering a single detector this happens for $\beta > 0.25$.

Estimations at measurement sites A and C for the same diameters show a similar behavior with small variations in precision; this was expected since the profiles change in time and space. For example, in the estimation for the obstacle with $\beta = 0.0122$, R^2 decreases approximately 16%. This could imply that measuring far away from the obstacle can still produce good estimations of the obstacle's size. Considering the results obtained in this case, it is not possible to establish for which of the physical variables (v_x or q) the ANN shows a better performance.

Figure 7 shows the results with a single sensor at $y = 0.21$ m, where we obtained values of R^2 above 0.8 considering v_x. This accuracy decreases when q is considered as input, obtaining the worst prediction at $R^2 = 0.231$. In the case with three sensors, the results improved considerably, maintaining a very similar behavior for both physical variables; the worst prediction obtained was $R^2 = 0.669$ for $\beta = 0.183$. Furthermore, in case 1c, $R^2 > 0.9$ for $\beta > 0.25$.

Following the approach in case 2a, where the ANNs are trained with the profiles of v_x or q measured at site B, with the intention of obtaining not only an estimation of the obstruction's size, but also its location on the y axis. As shown in Fig. 8, the results have an R^2 coefficient close to 1.0 for the larger obstacles ($\beta \geqslant 0.488$). However, in relation to the obstacle with $\beta = 0.366$, for which the ANN was not trained at all, the worst results decreased to a value of $R^2 = 0.356$ for v_x and $R^2 = 0.243$ for q. Note that both estimations are made with the obstacle near the walls of the pipeline. Meanwhile, the R^2 coefficient for the obstacle with diameter $\beta = 0.244$ has its lowest value when it is located at the center of the pipe, with $R^2 = 0.66$ and $R^2 = -0.033$ using the profiles of v_x and q, respectively. An example of a bad prediction from this case is shown in Fig. 9, where the target obstacle has a diameter of $\beta = 0.122$ and is located at $y = 0.35$ m; however the ANN
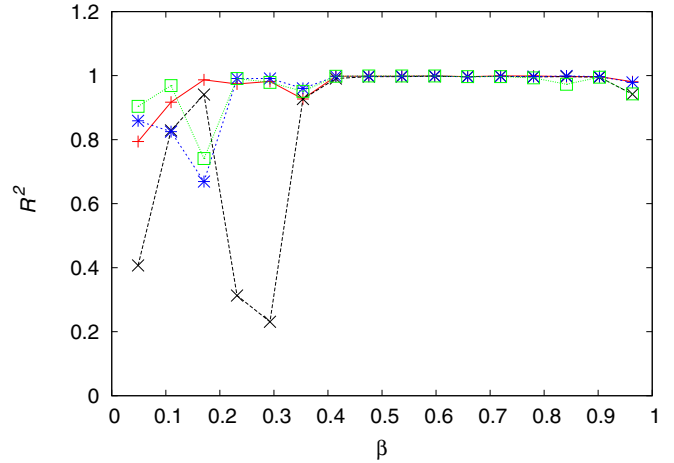
shows two obstacles, one at the top of the pipe, as expected, and small one at the bottom of the pipe.

Regarding case 2b, where the ANN is trained with the time series of a single sensor at the center of the pipe. The ANN is unable to learn the behavior of the flow, mostly for obstacles close to the borders as well as for small ones, as seen also in case 2a. This is evident in Fig. 10, where we get negative values for the small obstacles, $\beta = 0.122$, with a minimum value of $R^2 = -0.312$ when the obstacle is at the center of
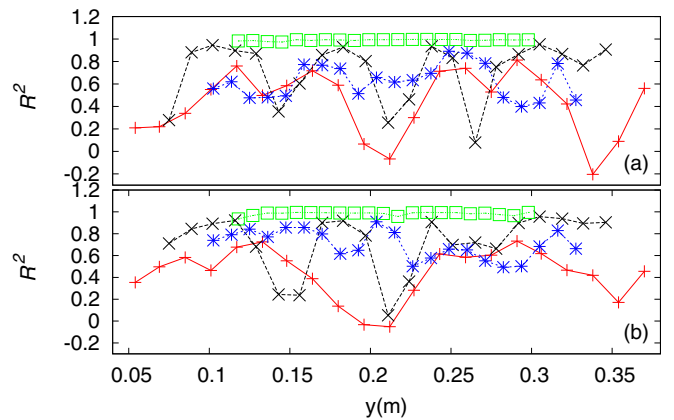

FIG. 8. Prediction analysis for obstacles in different positions on the y axis, by training and testing the ANN with the profiles of (a) v_x and (b) q for detectors at measurement site B. The red curve with crosses shows the results for the obstacle with $\beta = 0.122$; the black curve with X's, $\beta = 0.244$; the blue curve with asterisks, $\beta = 0.366$; and the green curve with squares, $\beta = 0.488$. Values of R^2 are similar for both approaches; the smallest obstacle has the worst prediction at the center and close to the borders of the pipe.
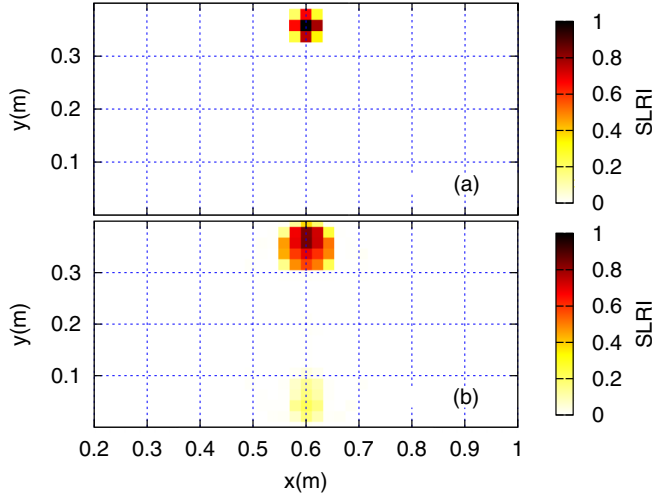
FIG. 9. (a) Target and (b) ANN prediction considering the profile of the v_x of the fluid generated by an obstacle at $y = 0.35$ m with $\beta = 0.122$. Contrasting with Fig. 6, we observe how the ANN shows two obstacles instead of a single one, resulting in bad prediction with $R^2 = 0.09$.

the pipe. For larger obstacles ($\beta \geqslant 0.244$) the ANN is more accurate, except when they are near the walls.

In case 2c, where the time series was generated with three equidistant detectors at $y = 0.105, 0.210$, and 0.315 m at measurement site B, the predictions shown in Fig. 11 indicate similar values of R^2 for both q and v_x. In this scenario, the predictions for the smallest obstacle ($\beta = 0.122$) show an improvement over their counterpart results for case 2b (Fig. 10). For the obstacle with $\beta = 0.366$ all values of R^2 are above 0.4 using v_x or q. Meanwhile, for the obstacles with $\beta = 0.244$ and $\beta = 0.488$ the prediction grid shows a great fit with respect to the target grid at almost all locations, with R^2
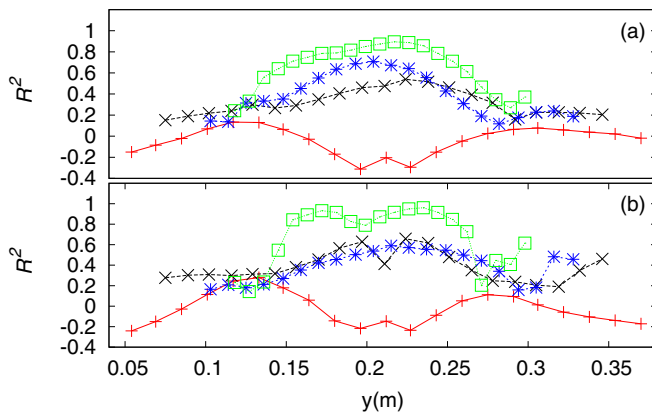

FIG. 10. Prediction analysis for obstacles in different positions on the y axis. The ANNs are trained with the time series of (a) v_x and (b) q at the center of the pipe at measurement site B. The red curve with crosses shows the results for the obstacle with $\beta = 0.122$; the black curve with X's, $\beta = 0.244$; the blue curve with asterisks, $\beta = 0.366$; and the green curve with squares, $\beta = 0.488$. It is evident that with the ANN it is difficult to give a proper prediction of the location of the obstacle with only one sensor; this is more evident for small obstacles close to the borders of the pipe.
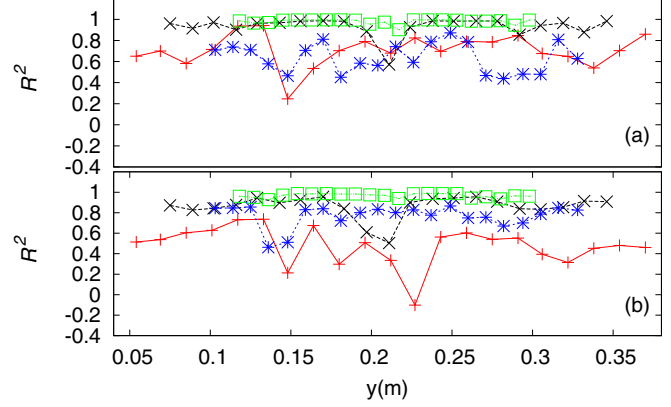

FIG. 11. R^2 results obtained for obstacles in different positions at the y axis. The ANNs are trained with the time series of (a) v_x and (b) q for three sensors at $y = 0.105, 0.210$, and 0.315 m at measurement site B. The red curve with crosses shows the results for the obstacle with $\beta = 0.122$; the black curve with X's, $\beta = 0.244$; the blue curve with asterisks, $\beta = 0.366$; and the green curve with squares, $\beta = 0.488$. Again, the worst adjustment coefficient was obtained with the smallest obstacle, $R^2 = -0.102$, using q as input data. However, there is a remarkable improvement for the other three obstacle sizes compared to the case where a single sensor was used.

above 0.8. However, the obstacle with $\beta = 0.122$ and located at the center of the pipe has a value of $R^2 = -0.102$ using the profile of q.

In general for case 2, the worst adjustment is obtained for the smaller obstacles. This confusion of the ANNs is not

TABLE III. R^2 for the test set, considering 10 values of the incoming fluid flow profile of v_x or q before the obstacle and 10 values at site B. Excluding the three smallest obstacles, the other results are prominent, with values close to 1.

| β	v_c (m/s)	R^2	
		For v_x	For q
0.0122	0.3	-23.753	-246.954
0.0122	1.35	-0.645	-0.680
0.0976	0.15	0.250	-4.518
0.0976	0.75	0.966	0.919
0.0976	1.8	0.975	0.975
0.1952	1.5	0.961	-0.016
0.2928	0.45	0.989	0.936
0.2928	1.65	0.989	0.944
0.3904	0.45	0.998	0.978
0.3904	1.2	0.991	0.997
0.4880	0.9	0.999	0.988
0.4880	1.05	0.998	0.994
0.4880	1.5	0.999	0.986
0.5856	1.05	0.998	0.997
0.5856	1.2	0.997	0.997
0.7808	0.75	0.999	0.996
0.7808	1.2	0.999	0.999
0.8784	1.35	0.999	0.992
0.9638	0.6	0.999	0.966
0.9638	0.9	0.999	0.992
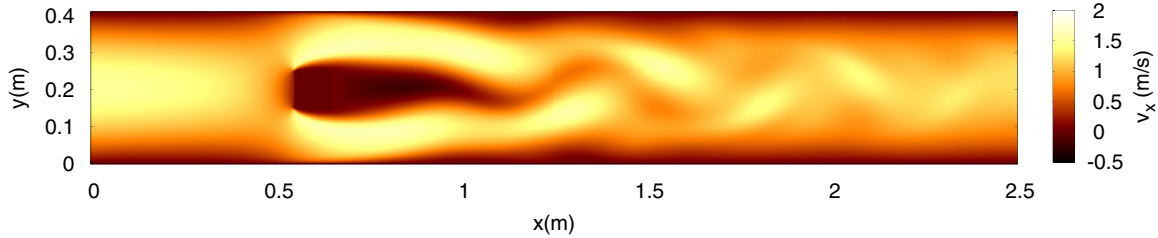
FIG. 12. Magnitude of v_x, for a simulation of a flow around a square obstacle with an incoming flow of $v_c = 1.5$ m/s. The obstacle is located at 0.6 m on the x axis. As with the cylindrical obstacle, the vortices formed after the square obstacle are driven along the direction of the flow, generating also a Karman vortex street, but with a different frequency.

surprising, since the pipe boundaries disturb the fluid flow generated after the obstacle even more when it is located near the walls rather than at the center. The boundary layer is affected by the viscous nature of the flow, and if the obstacle is very small, its effect on the flow is counteracted by viscous forces, which implies that when we measure far from the obstacle (for example, at measurement site B), the flow practically behaves as if there were no obstacle. Furthermore, smaller obstacles, regardless of their distance from the pipe boundaries, do not significantly affect the pattern of the flow that is generated behind them, so that the ANN cannot characterize them correctly. As seen in case 2b, a single sensor is not enough to obtain a good estimate and also reduces the accuracy with respect to that in case 2a. By increasing to three sensors in case 2c, we obtained a considerable improvement in terms of R^2 values.

The results in cases 1 and 2 show that as the number of measurement points in both space and time increases, the obstacle size prediction of the ANN improves considerably. For example, in case 1a, where the ANNs use 83 values in space and only 1 in time, the accuracy is equivalent to that in case 1c, where three measurements in space and 300 in time are used. A similar conclusion can be achieved by comparing cases 2a and 2c. In other words, the ANN in case 1a performs better than in case 1c, since case 1 requires fewer measurements over time. However, in a practical sense, having a measurement site with only 3 sensors can be more desirable than the approach of constructing a measurement site with more than 83 sensors.

For case 3, the R^2 coefficients obtained for each scenario are listed in Table III. Here we observe the persistent problem for the smallest obstacle ($\beta = 0.0122$), with an $R^2 = -23.753$ when v_x is used or even worse for q, with $R^2 = -246.954$. However, these particular results also have the second lowest incoming flow velocity, at $v_c = 0.3$ m/s. To understand this, compare the results obtained in case 1, where the associated v_c is always equal to 1.5 m/s. That is, a decrease in accuracy not only is associated with the difficulty of the ANN in characterizing the flow patterns for tiny obstacles, but also occurs because the velocity field of the flow around the obstacle is very small. In other words, in case 3 we found that, having both very small obstacles and incoming flow velocities, the ratios of inertial to viscous forces within the fluid are very low. Note the improvement of the results for the same value of $\beta = 0.0122$ with $v_c = 1.35$ m/s, obtaining $R^2 = -0.645$ for v_x or $R^2 = -0.680$ for q. The same happens for the obstacle with $\beta = 0.0976$, improving from $R^2 = 0.250$ to

$R^2 = 0.975$ and from $R^2 = -4.518$ to $R^2 = 0.975$ for v_x and q, respectively, when the incoming flow velocity is increased from $v_c = 0.15$ m/s to $v_c = 0.75$ m/s for both variables. This means that the ANN learns better as we increase the incoming fluid flow velocity. In general the accuracy of the predictions increases when v_x is used instead of q. It is noteworthy that for most of the remaining predictions the values of R^2 are close to 1, with a performance similar to that in case 1. Let us remark that this shows the flexibility of the ANN, which works with different initial flow velocities and employing only 10 sensors at both measurement sites.

We have proved that the ANNs achieve a good performance no matter whether we select the training, validation, and prediction sets in an orderly manner, as in cases 1 and 2, or randomly, as in case 3. The relevance of the latter case is that the ANN is trained not only for different diameters, but also for different incoming flow velocities, implying more complexity with respect to input information, resulting in a clear improvement in the recognition of the shape of obstacles. Despite this, bad results were obtained for $\beta = 0.0122$; the ANN was able to obtain $R^2 > 0.96$ for $\beta \geqslant 0.0976$ when using v_x as input, except in the particular case of $\beta = 0.0976$
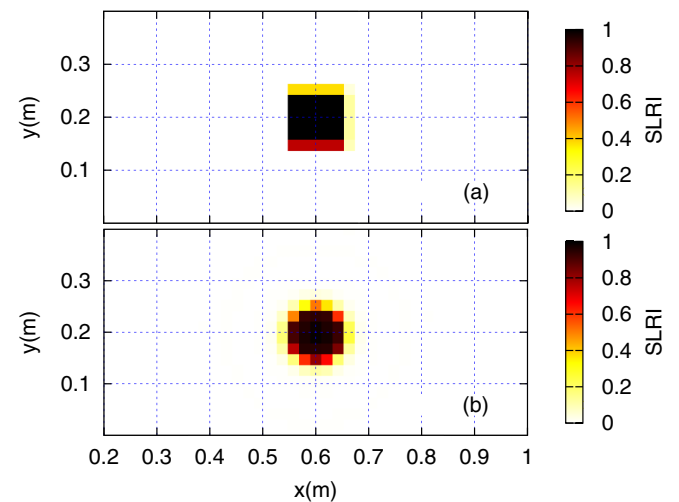

FIG. 13. Prediction for a square obstacle at the center of the diameter of the pipe, considering as input for the ANN the time series of q at three sensors at site B. As expected, the ANN considers this obstacle with a shape similar to those which it was trained for, however, it has a size and location comparable to those of the target.

with the low incoming velocity of $v_c = 0.15$ m/s, which is 10 times lower than the v_c used in case 1. A similar result is obtained when q is used, obtaining, in general, values of $R^2 > 0.915$. Note that, compared with case 1, the best results were obtained for $\beta > 0.25$ with $R^2 > 0.9$ for both v_x and q.

Hitherto, we have made predictions of the sizes and locations of the obstacles. But we can also test the ability of an already trained ANN to estimate the shape and size of a different obstacle. To show this capacity, we analyze a final experiment, introducing a square obstacle of side 0.1 m under the same conditions as presented in case 1 (see Fig. 12). For simplicity, we only present the case for the time series with three sensors at measurement site B (as in case 1). Although the predicted shape is like the ones for which the ANN was trained, we obtained a size similar to that of the square target (see Fig. 13), with the outstanding value of $R^2 = 0.93$.

V. CONCLUSIONS

A series of ANNs has been constructed and trained in the capability to estimate an obstacle's size and location inside a pipe with a specific width of 0.41 m, $1/80 \le \beta < 1$, and a range of incoming flows with characteristic velocities from 0.15 to 1.8 m/s. The ANNs use as input the profile of v_x or q at a certain distance from the obstacle. We analyzed several cases, varying the diameter (case 1), the position of the obstacle with respect to the y axis (case 2), and the incoming fluid velocity (case 3).

Based on the specifications used in this work, from the results obtained in case 1, the ANN is highly capable of generating estimations of the obstacle's size when the data supplied are very similar to those used in the training phase, with results of $R^2 > 0.9$ for values of $\beta > 0.25$. In case 2 something similar was done, considering an obstacle for which the ANN was not trained with any information about the profile or time series of v_x or q. Even when there was an evident decrease in accuracy, the ANN was able to estimate not only the shape but also the location of this obstacle, with an $R^2 > 0.6$

in general, and remarkable results for values of $\beta \geqslant 0.244$ when the obstacles are not too close to the walls of the conduit. Finally, in case 3 we have used multiple incoming flow velocities for each blockage considered, determining that for low velocities and small obstacles, the ANN predictions have a poor accuracy, while for bigger sizes and higher incoming flow velocities, the accuracy is improved considerably, achieving values of $R^2 > 0.9$ in general for $\beta \geqslant 0.0976$.

We found that the ANNs perform better in two situations: first, when the number of sensors at a measurement site is large as in case 1; and second, when the time series with three sensors is considered as in case 1. Nevertheless, we think that, in a practical sense, it is more convenient to extract data over a lapse of time with a few sensors. In the same context, the results obtained by training the ANNs with v_x or q are similar.

Finally, of all cases reviewed the best results are obtained in case 3, given that the ANN can give predictions with different incoming fluid velocities, while cases 1 and 2 were trained under only a single incoming fluid velocity. Furthermore, case 3 had fewer obstacles in the training set.

Taking into account that in the literature one can find works like [39], where they use modal analysis to identify a blockage through its location, thickness, and depth, and the study done in Ref. [40], where a transient pressure-wave reflection analysis is used to characterize the blockage inside the pipe and for more complex configurations such as the one described as motivation in Ref. [41], we would like to extend our research to develop a numerical tool capable of recognizing the shape and depth (position around the y axis) of such obstructions.

[1] R. Ugarelli, G. Venkatesh, H. Brattebø, V. Di Federico, and S. Saegrov, Urban Water J. **7**, 335 (2010).

[2] W. N. Kernan *et al.*, Stroke **45**, 2160 (2014).

[3] C. E. Davies and A. Desai, Powder Technol. **183**, 436 (2008).

[4] E. D. Sloan, Fluid Phase Equil. **228**, 67 (2005).

[5] F. Y. Fraige and P. A. Langston, Granul. Matter **8**, 67 (2006).

[6] S. P. Datta, P. K. Das, and S. Mukhopadhyay, Appl. Therm. Eng. **70**, 925 (2014).

[7] H. Wang, J. Tian, H. Ouyang, Y. D. Wu, and Z. H. Du, Int. J. Refrig. **46**, 173 (2014).

[8] H. F. Duan, J. Water Resourc. Plan. Manage. **142**, 04015073 (2016).

[9] R. M. Ashley, A. Fraser, R. Burrows, and J. Blanksby, Urban Water **2**, 263 (2000).

[10] J. Bailey, E. Keedwell, S. Djordjevic, Z. Kapelan, C. Burton, and E. Harris, Proc. Eng. **119**, 1288 (2015).

[11] J. P. Rodriguez, N. McIntyre, M. Diaz-Granados, and C. Maksimovic, Water Res. **46**, 4571 (2012).

[12] E. Deloose, P. Janssen, I. Depoortere, and J. Tack, Nature Rev. Gastroenterol. Hepatol. **9**, 271 (2012).

[13] C. J. Shi and R. H. Hruban, Hum. Pathol. **43**, 1 (2012).

[14] J. J. Chiu and S. Chien, Physiol. Rev. **91**, 327 (2011).

[15] K. Fujimoto, H. Sugiyama, and S. Yazaki, Cardiol. Young **25**, 731 (2015).

[16] W. Duan, R. Kirby, J. Prisutova, and K. Horoshenkov, Appl. Acoust. **87**, 190 (2015).

[17] J. Ma, M. J. S. Lowe, and F. Simonetti, Meas. Sci. Technol. **18**, 2629 (2007).

[18] N. L. T. Lile, M. H. M. Jaafar, M. R. Roslan, and M. S. M. Azmi, Int. J. Adv. Sci. Eng. Info. Technol. **2**, 252 (2012).

[19] C. Massari, T. C. J. Yeh, M. Ferrante, B. Brunone, and S. Meniconi, J. Hydroinformat. **16**, 248 (2014).

[20] M. Al-Naser, M. Elshafei, and A. Al-Sarkhi, J. Petrol. Sci. Eng. **145**, 548 (2016).

[21] E. S. Rosa, R. M. Salgado, T. Ohishi, and N. Mastelari, Int. J. Multiphase Flow **36**, 738 (2010).

[22] J. Ling and J. Templeton, Phys. Fluids **27**, 085103 (2015).

[23] I. Bright, G. Lin, and J. N. Kutz, Phys. Fluids **25**, 127102 (2013).

[24] P. Lauret, F. Heymes, L. Aprin, A. Johannet, and P. Slangen, Chem. Eng. Trans. **43**, 1621 (2015).

[25] O. Duran, K. Althoefer, and L. D. Senevatne, IEEE Trans. Automat. Sci. Eng. **4**, 118 (2007).

[26] A. A. Mohamad, *Lattice Boltzmann Method: Fundamentals and Engineering Applications with Computer Codes* (Springer, London, 2011).

[27] S. Chen and G. D. Doolen, Annu. Rev. Fluid Mech. **30**, 329 (1998).

[28] M. Schfer, S. Turek, F. Durst, E. Krause, and R. Rannacher, in *Flow Simulation with High-Performance Computers II: DFG Priority Research Programme Results 1993–1995*, edited by E. H. Hirschel (Vieweg+Teubner Verlag, Wiesbaden, 1996), p. 547.

[29] M. Carrillo, U. Que, and J. A. Gonzalez, Phys. Rev. E **94**, 063304 (2016).

[30] C. X. Pan, L. S. Luo, and C. T. Miller, Comput. Fluids **35**, 898 (2006).

[31] C. K. Aidun and J. R. Clausen, Annu. Rev. Fluid Mech. **42**, 439 (2010).

[32] D. Yu, R. Mei, L.-S. Luo, and W. Shyy, Prog. Aerospace Sci. **39**, 329 (2003).

[33] M. Bouzidi, M. Firdaouss, and P. Lallemand, Phys. Fluids **13**, 3452 (2001).

[34] Z. G. Feng and E. E. Michaelides, J. Comput. Phys. **195**, 602 (2004).

[35] D. Yu, R. Mei, and W. Shyy, Prog. Comput. Fluid Dynam. **5**, 3 (2005).

[36] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, Nature **323**, 533 (1986).

[37] C. Bishop, *Machine Learning for Pattern Recognition* (Springer-Verlag, New York, 2006).

[38] R. Rojas, *Neural Networks: A Systematic Introduction* (Springer-Verlag, Berlin, 1996).

[39] O. Bello, N. Virani, and S. O. Oyadiji, in *ASME 2011 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference* (American Society of Mechanical Engineers, New York, 2011), Vol. 1, pp. 551–559, paper no. DETC2011-48787.

[40] N. Adeleke, M. T. Ityokumbul, and M. Adewumi, SPE J. **18**, 355 (2013).

[41] Z. M. Yuan, Z. B. Deng, M. Z. Jiang, Y. Xie, and Y. Wu, J. Nat. Gas Sci. Eng. **22**, 141 (2015).