Communications in Science and Technology 9(2) (2024) 291-301

COMMUNICATIONS IN SCIENCE AND TECHNOLOGY
Homepage: cst.kipmi.or.id

Data-driven modeling approaches for pressure drop prediction in a multi-phase flow system

Nezar M. Alyazidi*[1,2], Aiman F. Bawazir[1], Ala S. AL-Dogail[3]

[1]*Control and Instrumentation Engineering Department, King Fahd University of Petroleum & Minerals, Dhahran, 31261 Saudi Arabia*
[2]*Interdisciplinary Center of Smart Mobility and Logistics, King Fahd University of Petroleum & Minerals, Dhahran, 31261 Saudi Arabia*
[3]*Petroleum Engineering Department, King Fahd University of Petroleum & Minerals, Dhahran, 31261, Saudi Arabia*

Article history:
Received: 6 May 2024 / Received in revised form: 14 November 2024 / Accepted: 29 November 2024

Abstract

Accurate prediction of pressure drops in multi-phase flow systems is essential for optimizing processes in industries such as oil and gas, where operational efficiency and safety depend on reliable modeling. Traditional models often need help with the complexities of multi-phase flow dynamics, resulting in high relative errors, particularly under varying flow regimes. In this study, we simulate a comprehensive multiphase flow experimental data collected from the lab. This study presents innovative methods for accurately modeling pressure drops in multi-phase flow systems. It also studies the complicated dynamics of multi-phase flows, which are flows with more than one phase at the same time. It does this by using two different data-driven models, nonlinear ARX and Hammerstein-Wiener, instead of neural networks (NNs), so that the models don't get too good at fitting environments with lots of changes and little data. Our research applies system identification approaches to the intricacies of this domain, providing new insights into choosing the best appropriate modeling strategy for multi-phase flow systems, taking into account their distinct properties. The experimental results show that the nonlinear Hammerstein-Wiener and ARX models were much better than other methods, with fitting accuracy rates of 81.12% for the Hammerstein-Wiener model and 86.52% for the ARX model. This study helps the creation of more advanced control algorithms by providing a reliable way to guess when the pressure drops and showing how to choose a model that fits the properties of the multi-phase flow. These findings contribute to enhanced pressure management and optimization strategies, setting a foundation for future studies on real-time flow control and broader industrial applications.

Keywords: Data driven; multi-phase modeling; pressure drop; neural network; nonlinear autoregressive exogenous; Hammerstein model

1. Introduction

The dynamics of multi-phase systems are more complex than those of single-phase systems because of the differences in phase distribution, resulting in unique flow patterns or regimes [1]. Several variables influence the flow patterns and pressure drops that occur at specific points in the pipe. Those variables include turbulence, surface tension forces, inertia, pipe diameter, and volume fluxes. The flow pattern maps and equations exhibit insufficient accuracy, with a high relative error ranging from 20 to 30 percent [2]. This underscores the necessity for an accurate and reliable method to identify the flow patterns and pressure drop. Researchers widely use correlations from laboratory experiments, particularly [3], [4], [5], and others. Numerous studies have examined the issues with these multi-phase flow correlations, concluding that they are only effective within a specific range of input parameters.

The process dynamics, system response, and system stability, including the step response and poles and zeros, have been analyzed using a data experiment [6]. Additionally, [7] used system identification derived from data collected in an Excel spreadsheet to characterize the dynamic behavior of a system. This model was created using a number of approaches, such as OKID-based identification [8,9], Hammerstein-Wiener [10], and ARX identification [11].

The review of the literature reveals the utilization of machine learning (ML) algorithms in predicting pressure gradients and directly estimating pressure outputs. Al-Naser et al. [12] came

*Corresponding author.
Email: nalyazidi@kfupm.edu.sa
https://doi.org10.21924/cst.9.2.2024.1430

up with a new way to predict flow patterns in horizontal pipes. It used artificial neural networks (ANNs) and three dimensionless parameters: the liquid Reynolds number, the gas Reynolds number, and the pressure drop multiplier. These parameters made the model more practical for a wider range of situations. They also employed a preprocessing phase that involved natural logarithmic normalization of a large dataset, resulting in significant improvements in the accuracy of the predictions. A study by Najafi [13] created a way to use machine learning to make better predictions about frictional pressure drop for both single-phase (water) and two-phase (water-air) flows in micro-finned tubes. This method was much more accurate than traditional physical models. In a different study, Al-dogail [14] created dimensionless machine learning models to predict the drop in pressure in multi-phase flow through horizontal pipes. A wider range of design and operation situations can utilize these more accurate models. By incorporating key fluid properties—such as density, viscosity, and surface tension—the models achieve high prediction accuracy with minimal inputs.

Artificial neural networks (ANN) are widely used to estimate pressure drops. As shown in [15], ANN was used to estimate the pressure drop of the oil-water-air mix moving through both horizontal and vertical tubes. This technique was used to forecast the pressure reduction in R407C flow during evaporation within horizontal smooth tubes [16]. It has been used recently to figure out the pressure drop in CuO/(ethylene glycol-water) nanofluid flows in car radiators [17] and in non-azeotropic mixes going through cryogenic forced boiling [18]. ANN and reinforcement learning (RL) were also used together to manage the nonlinear dynamics of a multi-phase system dynamics model based on experimental data [19].

In a separate study, Shanthi and Pappa aimed to improve the precision of identifying flow patterns in two phases using artificial intelligence techniques that rely on the characteristics extracted from images of the flowing material. They employed fuzzy logic and support vector machines (SVMs) with principal component analysis (PCA) to accomplish this goal [20]. In their study, Xiao et al. [21] created an AI system that uses fuzzy logic, support vector machines, and principal component analysis to predict how gases and liquid nanofluids will flow in vertical mini-channels. In [22], the research employed both experimental observation and a two-dimensional numerical model to show that effective management of the water table can achieve targeted soil moisture distribution, soil pressure, and flux.

Although machine learning techniques, such as artificial neural networks (ANNs), have shown promise in addressing these challenges, they often need more generalization across varying flow regimes, limiting their real-world applicability. To overcome these limitations, this study introduces a hybrid data-driven approach that leverages nonlinear ARX and Hammerstein-Wiener models, offering enhanced adaptability and precision for pressure drop predictions in multi-phase flow systems. This approach accurately captures complex system dynamics and provides a robust foundation for advanced control and optimization strategies. This work represents an initial endeavor to apply these data-driven techniques specifically for pressure prediction in multi-phase systems, emphasizing the urgency for more precise and adaptable models in industrial applications. The contributions of this study are as follows:

- We conducted a comprehensive experiment to collect multi-phase flow data in a laboratory environment. We then use several data-driven techniques to predict the pressure level in multi-phase flow systems.
- We utilized neural networks, nonlinear Hammerstein-Wiener, and nonlinear ARX models to determine the most effective model for predicting the pressure level based on the experimental data.
- This study shows new ways to model how pressure drops in multi-phase flow systems accurately.

2. Materials and Methods

In the context of multi-phase flow dynamics, both the flow pattern and the associated pressure gradient exhibit significant variability along the pipeline's length, necessitating its segmentation into discrete sections for analytical precision [23]. Within each segment, the flow regime remains uniform and is characterized by a relatively stable pressure gradient. Utilizing numerical methodologies, the pressure gradient in each segment is systematically computed by applying established multi-phase flow correlations and mechanistic modeling frameworks.

As detailed in [1], an experimental apparatus comprising a horizontal flow loop with a diameter of 0.0254 m (1 inch) and a total length of 9.15 m (30 ft) was employed to examine flow regime transitions and pressure drop behavior. This flow loop features a steel reservoir, a PVC pipe segment measuring 0.0254 m (1 inch) in diameter and 8 m (27 ft) in length, complemented by a 1 m (3 ft) section of plexiglass pipe. The system is further equipped with a centrifugal pump integrated with a variable speed drive (VSD), an air compressor paired with a dryer, dual liquid flow meters, dual gas flow meters, two pressure gauges, a differential pressure gauge, and a high-resolution imaging system, as schematically illustrated in Fig. 1.

Surface tension was systematically modified by introducing a surfactant into the water phase, while density was adjusted using calcium bromide, and viscosity was tailored through the addition of glycerin. The experimental conditions encompassed a range of superficial gas velocities spanning from 0 to 18.288 m/s (0–60 ft/s) and superficial liquid velocities varying from 0 to 3.048 m/s (0–10 ft/s). To evaluate the hysteresis effect, the gas velocity was incrementally increased from its minimum to maximum values, followed by a decremental reversal from maximum to minimum, with the liquid flow rate held constant throughout the tests.

At each stage of the experiment, steady-state conditions were ascertained based on the stabilization of pressure responses and flow rates. Data acquisition was conducted with meticulous attention to ensure sufficient data points for robust analysis. To enhance the understanding of the integration and functionality of sensors and actuators within this investigation, as well as to facilitate precise fluid dynamic measurements, the configuration of the experimental apparatus is illustrated in Fig. 1.

3. Results and Discussions

Capturing the intricate dynamics of complex systems necessitates a comprehensive approach that transcends superficial methodologies. This section underscores the critical importance of accurate modeling in control system design, advocating for a

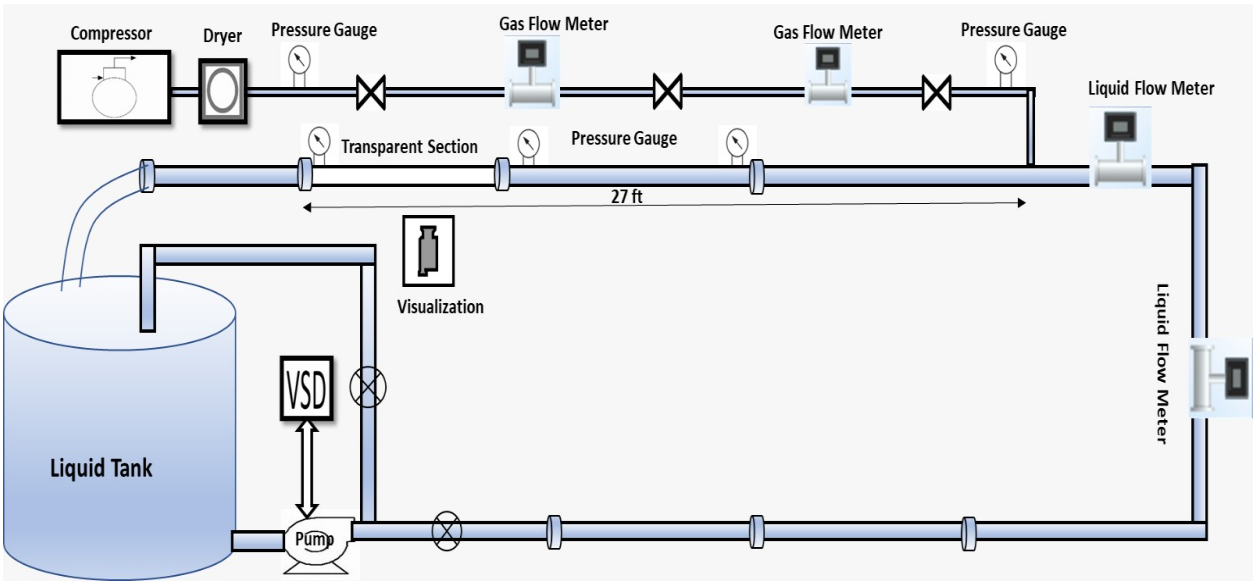

Fig. 1. Schematic of the experimental setup.

Table 1. Pipe Specification

Parameter	Value	Sample Code
Number of samples	1042	N.A.
Pipe diameter (inch)	1.00	D_{inch}
Pipe diameter (meter)	0.0254	D_{meter}
Pipe Length (ft)	30	L_{inch}
Pipe Length (meter)	9.15	L_{meter}

Table 2. Experimental Parameters

Parameter	Value	Sample Code
Gas superficial velocity	(m/s)	
-Min	0	v_g
-Max	18.288	
Liquid superficial velocity	(m/s)	
-Min	0	v_l
-Max	3.048	
Liquid flow rate	Liters/min	
-Min	0	Q_l
-Max	96.5	
Liquid Surface Tension	(mN/m)	
-Min	32.4	σ
- Max	70.5	
Liquid density	(kg/m^3)	
- Min	998.15058	ρ
- Max	1506.788153	
Liquid viscosity	(Pa.s) (x10^{-3})	
- Min	1.000	μ
- Max	3.10035936	

departure from traditional linear paradigms to address the inherent nonlinear complexities characteristic of real-world processes. A significant subset of chemical and industrial processes exhibits pronounced nonlinearity, necessitating the development of robust mathematical models for simulation and control purposes. Such models serve as formalized frameworks that delineate the intrinsic relationships between a system's inputs and its outputs.

The construction of process models can be pursued through two principal methodologies. Physical modeling involves the derivation of process models grounded in fundamental principles, encompassing the laws of physics, chemistry, and biology [24]. In contrast, process identification employs empirical data obtained through experimental testing to construct dynamic process models. Commonly referred to as black-box modeling, this approach eschews explicit reliance on physical principles, favoring data-driven methodologies. Compared to physical modeling, process identification offers several notable advantages:

- Cost-efficiency in model development: By leveraging empirical data, process identification avoids the labor-intensive requirements of deriving models from first principles, making it a practical and economical option across diverse applications.

- High precision within specific operational ranges: Identified models exhibit exceptional accuracy within the operational conditions tested, ensuring precise control and predictive capabilities in defined scenarios.

- Improved reliability for pressure drop predictions in multiphase systems: This research emphasizes enhancing the precision and reliability of pressure drop forecasts in industrial flow applications, supporting the design and optimization of systems critical to operational efficiency.

- Structural simplicity: Identified models are often less complex and easier to implement compared to physically derived counterparts, while still capturing essential system dynamics effectively. However, a notable limitation of process identification lies in its applicability: the derived models are valid only within the tested operational range, potentially restricting their generalizability.

System identification is a pivotal scientific methodology employed to analyze and quantify data from a system, facilitating

the construction of mathematical models that accurately represent system behavior. This approach enables researchers and engineers to uncover the underlying physical and mathematical principles governing complex systems, empowering them to make informed predictions and enhance system performance. System identification finds extensive application across domains such as control engineering, signal processing, and robotics, serving as a cornerstone for advancing scientific research and technological innovation.

Despite its strengths, many conventional identification methods assume that input data are devoid of noise, which is often not the case in practical scenarios. Measurement noise, particularly when present in both input and output data, can compromise the validity of parameter estimates for identified models. This limitation is addressed by the errors-in-variables (EIV) model, which accounts for noise in both input and output variables [25], thus offering a more robust framework for reliable parameter estimation in noisy environments.

3.1. Neural Network-Based Model

In recent years, the adoption of neural networks across a diverse array of disciplines has expanded at an unprecedented rate, with model identification emerging as a domain of significant advancement. Conventional model identification techniques often depend on mathematical frameworks and assumptions that may inadequately represent the intricate behaviors of real-world systems. Neural networks, by contrast, offer a compelling alternative due to their ability to learn and generalize complex patterns directly from data. This section explores the development and application of a neural network-based approach to model identification, presenting the outcomes of its implementation in a practical context. Neural networks are particularly well-suited for approximating nonlinear functions with exceptional accuracy, as highlighted in [26]. The adjustment of neural network parameters enables the modeling of diverse nonlinearities, achieved through the application of a gradient descent optimization algorithm. This algorithm minimizes an error function that quantifies the difference between the predicted outputs of the neural network and the actual system outputs, given a set of input data or input-output data pairs (training data) [26, 27].

3.1.1. Methodology

- Data Collection: to train the neural network for model identification, a dataset comprising input-output pairs from the target system was collected. The dataset included five inputs and single output as well as a diverse range of operating conditions and disturbances to ensure the neural network's robustness.

- Neural Network Architecture A feedforward neural network structure was selected due to its simplicity and efficiency in capturing nonlinear connections. The architecture comprised an input layer, one or more hidden layers employing rectified linear unit (ReLU) activation functions, and an output layer. The determination of the number of neurons in each layer and the network depth was achieved through an iterative process of experimentation and optimization.

- Training Procedure: the neural network was trained using a supervised learning approach. The dataset was randomly split into training and validation sets to monitor the model's performance during training. The training process involved minimizing a suitable loss function through backpropagation and gradient descent optimization.

- Results and Discussion: MATLAB is employed to construct and train a feedforward neural network characterized by a three-layer architecture, comprising 32, 64, and 128 neurons, respectively. Each layer utilizes the hyperbolic tangent sigmoid transfer function to capture nonlinear relationships effectively. The network is trained using the scaled conjugate gradient descent algorithm (trainscg), with the Levenberg-Marquardt algorithm (trainlm) available as an alternative for enhanced optimization flexibility.

  The dataset is systematically partitioned into training, validation, and testing subsets in proportions of 70%, 15%, and 15%, respectively, ensuring a balanced approach to model evaluation and generalization. Critical training parameters, including the number of epochs, minimum gradient threshold, and maximum allowable validation failures, are precisely defined to optimize the training process.

  Upon completion of the training phase, the network is subjected to evaluation using the input dataset, and its performance is rigorously assessed. Additional functionalities, such as visualization of the network architecture, saving the trained model, and generating a corresponding Simulink model, are incorporated within the script. These features are included as commented-out sections, enabling users to activate them as required, thus providing a versatile and comprehensive framework for neural network development and deployment in MATLAB.

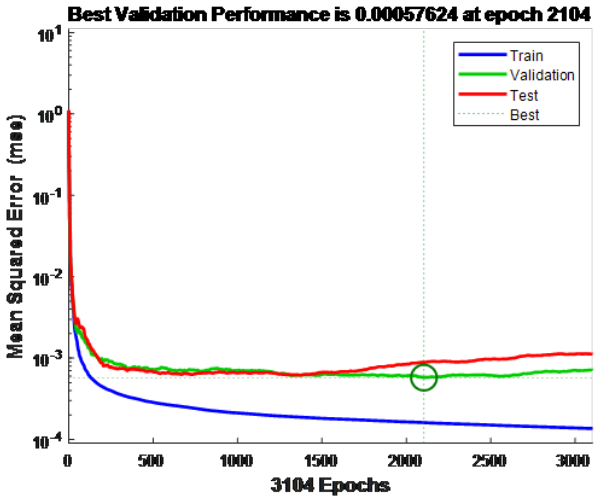

Fig. 2. Neural network training performance.

The neural network-based model identification method offers several advantages over traditional approaches. It can handle complex, nonlinear relationships without relying on explicit mathematical formulations, making it suitable for systems with unknown or highly nonlinear dynamics. Additionally, the ability of a neural network to learn from data enables the identification of intricate patterns that may be challenging to capture using conventional methods. Despite its advantages, the neural

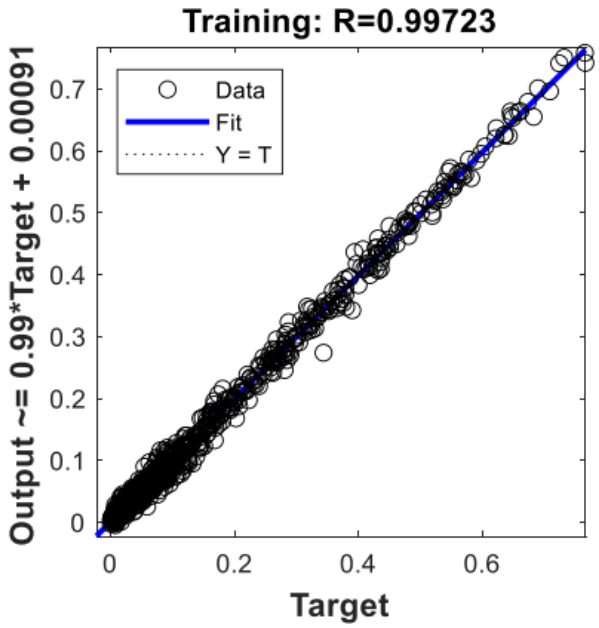

Fig. 3. Neural network training plot

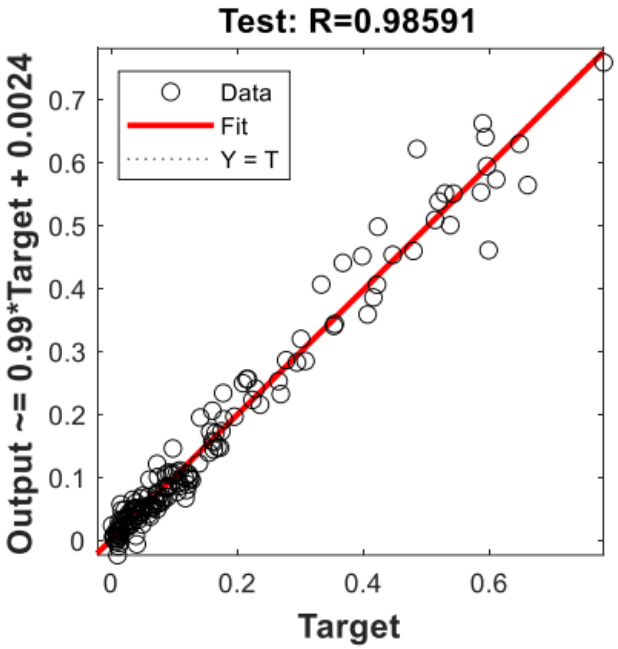

Fig. 5. Neural network test plot

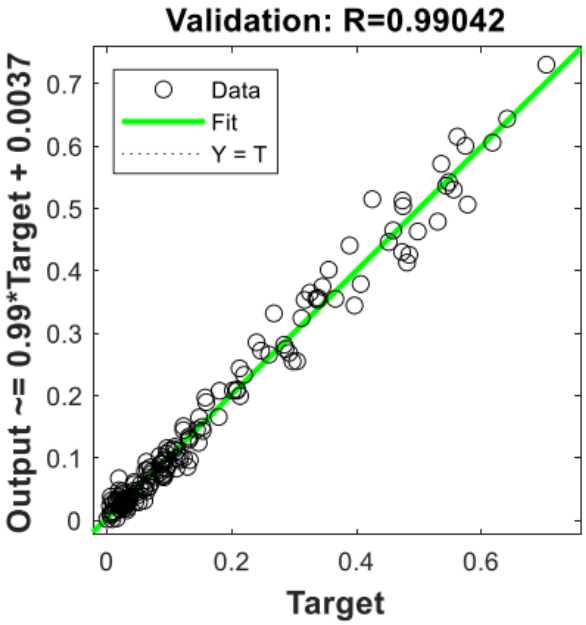

Fig. 4. Neural network validation plot

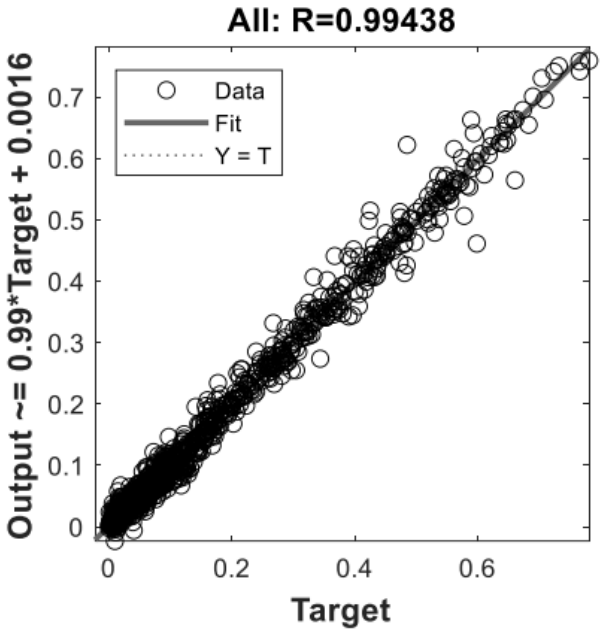

Fig. 6. Neural network all plot

network-based approach poses challenges such as the need for substantial amounts of training data and the potential for overfitting. Careful attention must be paid to the selection of network architecture and hyperparameters to ensure robust performance across various scenarios.

3.2. Nonlinear ARX Model

Nonlinear Auto-Regressive with Exogenous Inputs (ARX) models extend the capabilities of their linear counterparts to address the complexities of nonlinear systems. These models leverage a variety of functional forms, such as wavelet transformations and sigmoid networks, to effectively capture intricate nonlinear dynamics. Unlike conventional methods for nonlinear system identification, which often rely on rigidly defined model structures, nonlinear ARX models offer significant flexibility by supporting a range of adaptable configurations. For instance, these models can seamlessly integrate a hybrid framework that combines linear and nonlinear memoryless dynamical components, thereby enhancing their capacity to represent diverse system behaviors.

3.2.1. Nonlinear ARX Model Structure

A nonlinear (ARX) model is composed of model regressors and an output function. The output function integrates one or more

mapping objects, each associated with a model output. The schematic depicted in Fig. 7 outlines the architecture of a nonlinear ARX model. Each mapping object may encompass both linear and nonlinear functions, which act on the model regressors to generate the model output along with a fixed offset. This structure exemplifies the configuration of a single-output nonlinear ARX model within a simulation framework. This section introduces a novel methodology for nonlinear system identification utilizing operational data formatted within the ARX framework, wherein the parameters exhibit nonlinear dependencies on both input and output variables. Parameters for a given level of input and output are inferred from corresponding datasets. By iteratively performing identification across various input and output levels, the nonlinear dependencies of the parameters within the specified range can be systematically determined. In cases where the model order is unknown, a minimal realization of the locally identified model is performed, and the parameters are expressed as polynomial functions of the input and output variables, thereby constructing the nonlinear ARX model. This approach proves particularly advantageous for simplifying complex yet stable nonlinear systems into concise nonlinear ARX representations, which can subsequently be employed in the design of control systems [28]. This part outlines the proposed method for detecting

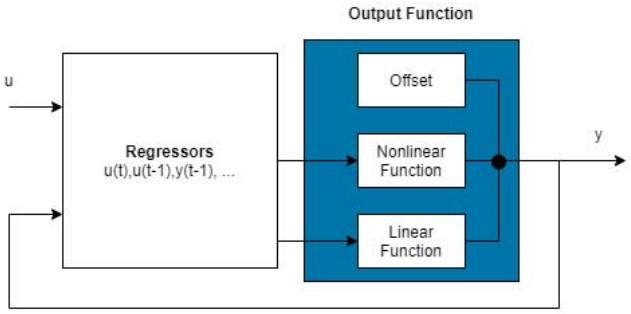

Fig. 7. Nonlinear ARX model structure.

a nonlinear ARX model where the parameters exhibit nonlinear relationships with both the input and output [28]. The equation representing the ARX model for the discrete-time linear system is given as follows:

$$y[k]+a_1y[k-1]+a_2y[k-2]+\ldots+a_ny[k-n]= b_1u[k-1]+b_2u[k-2]+\ldots+b_nu[k-n]+c \quad (1)$$

In the given equation, $y[k]$ represents the output and $u[k]$ represents the input at time k. The coefficients a_i and b_i (where $i=1,2,3,\ldots,n$) reflect the values assigned to the variables, while n indicates the system's order. The values of the parameters of the model are determined by the output $y[j]$ and the input $u[j]$ at time k, where j is equal to $k-n,k-n+1,k-n+2,\ldots,k$.

3.2.2. Dependency of ARX Parameters on Output

This subsection provides a comprehensive discussion of the identification technique for the nonlinear ARX model, with a specific focus on the output aspect. In this particular instance, the process of identifying the nonlinear system requires an iterative approach to determine the parameters in the ARX model. This is done by using input-output data from various output levels, as shown in Fig. 8. The ARX model obtained corresponds to the localized linear system centered around the selected output tier. The parameters assigned to different output tiers can be represented as functions that depend on the output. The process of determining the relevant local linear ARX model is illustrated in Fig. 8. The parameters are determined through the least squares criterion, defining the linear ARX model around y_o as:

$$\begin{aligned} &y[k_1]+a_1y[k_1-1]+\ldots+a_ny[k_1-n]=b_1u[k_1-1]\\ &+\ldots+b_nu[k_1-n]+c\\ &y[k_2]+a_1y[k_2-1]+\ldots+a_ny[k_2-n]=b_1u[k_2-1]\\ &+\ldots+b_nu[k_2-n]+c\\ &y[k_3]+a_1y[k_3-1]+\ldots+a_ny[k_3-n]=b_1u[k_3-1]\\ &+\ldots+b_nu[k_3-n]+c \end{aligned} \quad (2)$$

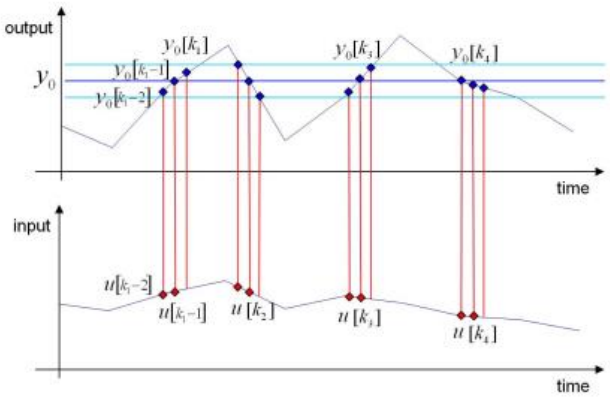

Fig. 8. Identification around the designated output level [29]

As indicated in equation 3, the coefficients of the ARX model are expressed as functions of the output y_o

$$y[k]+a_1(y_o)y[k-1]+\ldots+a_n(y_o)y[k-n]= b_1(y_o)u[k-1]+\ldots+b_n(y_o)u[k-n]+c(y_o) \quad (3)$$

To avoid using output data across multiple levels, the process of identifying a local ARX model can be carried out by choosing the output to stay within a specific range centred around a particular output level. The successful identification of the nonlinear ARX model is achievable through iterative identification across the defined output range and combining the obtained models.

3.2.3. ARX Model Parameters Variation with Input and Output

In the preceding section, we explored identifying ARX model parameters associated with the output. This section presents parameters influenced by the input variable (u) and the output variable (y) through an iterative process. This involves detecting local autoregressive exogenous (ARX) models where the input and output values fall within a predetermined range. Establishing an ARX model involves the methodical selection of various input and output levels, with the parameters affected by both the input and output variables being determined using equation 4.

$$y[k]+a_1(u,y)y[k-1]+\ldots+a_n(u,y)y[k-n]= b_1(u,y)u[k-1]+\ldots+b_n(u,y)u[k-n]+c(u,y) \quad (4)$$

The synthesis of local ARX models recognized at various stages of both input and output throughout the designated range achieves the determination of the nonlinear ARX model.

3.2.4. Results of Nonlinear ARX Model Identification

Following the preprocessing of the data, the dataset is utilized within MATLAB for system identification. This subsection presents the results derived from various experimental setups and discusses the effects of data loss and model fitting on the estimation process, thereby validating the proposed algorithm. Fig. 9 illustrates the results obtained using a nonlinear ARX model with exogenous inputs (nonlinear ARX) applied to time-domain data with a sampling interval of 1 second. The model configuration includes one output variable and five input variables. The regressors are composed of both linear terms and second-order polynomial terms, capturing the nonlinear dependencies within the data. The output function employs a support vector machine (SVM) with a Gaussian kernel to enhance the model's nonlinear approximation capability. The model achieved a fit of 78.77% to the estimation data, with a corresponding mean squared error (MSE) of 0.04504. These metrics highlight the efficacy of the proposed approach and its robustness in handling the complexities of system identification under the given constraints.

Fig. 10 was estimated using a Nonlinear ARX model of 5 inputs and one output on time domain data of sampling time 1 second. The regressors are linear regressors and polynomial order two regressors. The output function is a sigmoid network with 25 units. The fit to estimation data is 86.52%, and the mean square error is 0.01816.

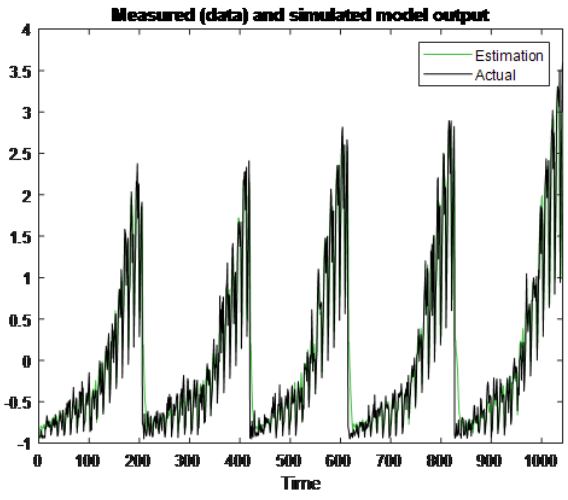

Fig. 9. Model estimation using NLARX model with support vector machine as output function

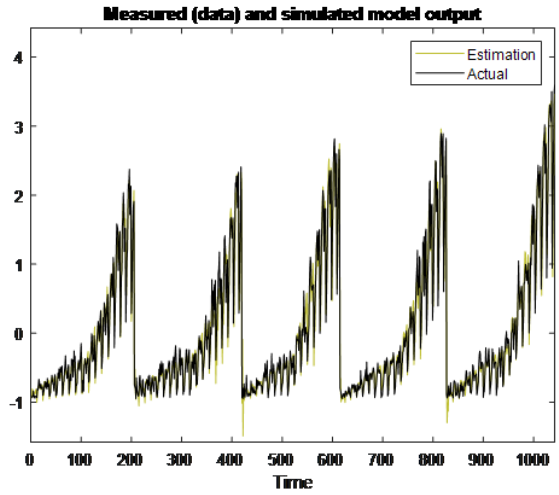

Fig. 10. Model estimation using NLARX model with the sigmoid network as output function

3.3. Nonlinear Hammerstein-Wiener Model

N-L-N models, also referred to as nonlinear block-oriented models, belong to a distinct category of models. These models incorporate a Linear (L) subsystem that is situated between two subsystems of static Nonlinearity (N) [30]. By incorporating two nonlinear components instead of just one, the modeling capabilities of H-W models are enhanced, particularly for nonlinear systems that display both actuator and sensor nonlinearities [31][32]. The recognition of H-W models has garnered significant interest in recent years, with multiple studies exploring various frameworks.

3.3.1. Nonlinear Hammerstein-Wiener Model Structure

The structure of the model is formulated by dividing the system into blocks. These blocks consist of static nonlinear components combined with dynamic linear terms to form the complete system. The H-W model operates in discrete-time. Fig. 11 illustrates the structural depiction of the H-W model. Specifically, if the model integrates solely the input nonlinearity f, it is denoted as a Hammerstein model (see Fig. 12). Likewise, if the model encompasses nonlinearity h solely at the output, it is termed a Wiener model.

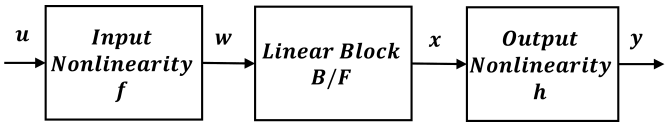

Fig. 11. Hammerstein-Wiener model structure

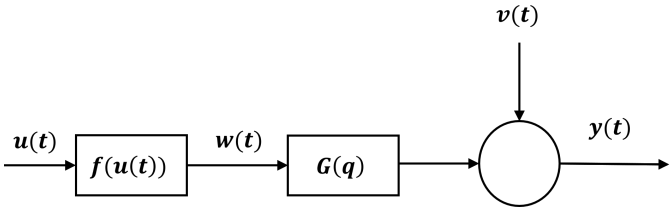

Fig. 12. The Hammerstein model

3.3.2. Results of Nonlinear Hammerstein Model Identification

In the context of identifying a Hammerstein model using MATLAB's System Identification Toolbox, the static nonlinearity component can be specified as five sigmoid networks (as the number of the inputs) with 10 units each. This configuration enables the model to capture intricate and non-linear connections between the input and output signals. The Levenberg-Marquardt (LM) searching method, a widely used optimization algorithm, is employed in the identification process. The LM approach enables the adjustment of the network's parameters, leading to enhanced

accuracy in approximating nonlinearities. The dynamic linear part is estimated using a transfer function with 3 poles and 4 zeros. The specific architecture and the LM searching method together enable a robust and efficient identification of the model. The fitting to estimation is 66.9%, and the mean square error is 0.1116. The representation of the underlying system dynamics is illustrated in the Fig. 13.

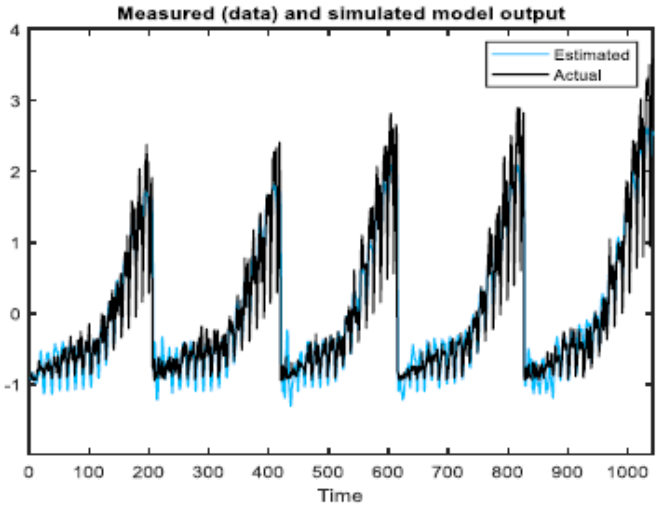

Fig. 13. Model estimation using nonlinear Hammerstein model with a sigmoid network of 10 units and order four TF

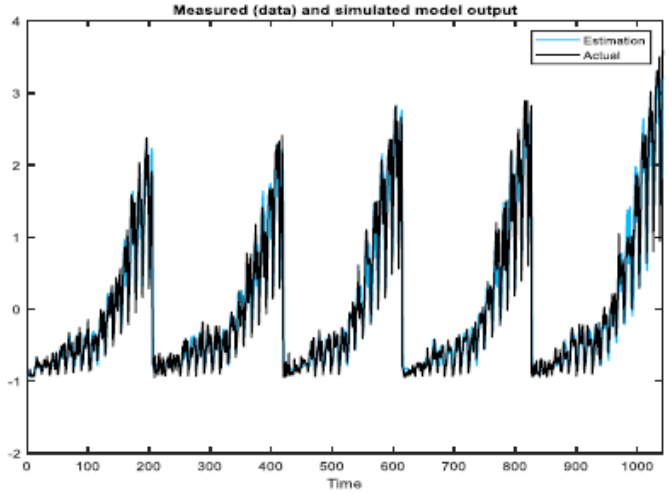

Fig. 14. Model estimation using nonlinear Hammerstein model with sigmoid network of 15 units and order 3TF

Repeating the same process and increasing the number of units in the static function to 15 and reducing the order of the linear dynamic function to 3, the fitting to estimation is increased to 80.12%, and the mean square error is reduced to 0.03956. See figure 14.

3.3.3. Identification of SISO Hammerstein Model

In this section, the methodology employed in the preceding analysis is applied to identify the Hammerstein model. This model characterizes a discrete-time, single-input, single-output (SISO) system that is linear and time-invariant. The system's input is defined as the ratio of liquid velocity to gas velocity, while the output corresponds to the observed pressure drop, expressed in psi.

The Hammerstein model consists of a sequential arrangement of static and dynamic components. Nonlinearities in the input are captured and transformed into an intermediate variable, $w(t)$, through the static input block, as formulated in Equation 6. Once transformed, the unmeasured intermediate variable $w(t)$ is processed by a linear functional block characterized by the time-invariant linear transfer function $G(q)$, which is responsible for converting the linearized input into the output, $x(t)$, as detailed in Equation 7 and described in [33]. A noise term, $v(t)$, modeled as a stationary, bounded stochastic signal, is incorporated as expressed in Equation 9.

It is critical to emphasize that the nonlinearity in this framework is confined to the input transformation; the model's output remains strictly linear. The degrees of the numerator and denominator polynomials of the linear transfer function $G(q)$ are denoted by the parameters m and n, respectively. Furthermore, the index α quantifies the extent of nonlinearity within the system.

The nonlinear function, which is parameterized as a polynomial, is formally expressed a:

$$f(u(t)) = \beta_1 u(t) + \beta_2 u^2(t) + \cdots + \beta_m u^m(t) \tag{5}$$

$$w(t) = f(\alpha, u) \tag{6}$$

$$x(t) = \frac{B(q^{-1})}{A(q^{-1})} w(t) \tag{7}$$

$$G(q) = \frac{B(q)}{A(q)} = \frac{b_0 + b_1 q^{-1} + \cdots + b_m q^m}{1 + a_1 q^{-1} + \cdots + a_n q^{-n}} \tag{8}$$

$$\begin{aligned} v(t) = H(q)e(t) &= \frac{1 + c_1 q^{-1} + \cdots + c_n q^{-l}}{1 + d_1 q^{-1} + \cdots + d_n q^{-l}} \\ &= \frac{C(q)}{D(q)} e(t) \end{aligned} \tag{9}$$

Upon applying these equations to the (SISO) Hammerstein model, the resulting structure is as follows:

$$y(t) = \frac{B(q)}{A(q)} \sum_{k=1}^{m} \beta_k u^k(t) + \frac{C(q)}{B(q)} e(t) \tag{10}$$

where:

- l denotes the linear component order in the Hammerstein model.
- q^{-1} is the delay operator symbolic representation and the complex variable z^{-1}.
- $H(q)$ represents the linear filter transfer function.
- $e(t)$ is a zero mean value and variance of λ^2 white noise sequence.
- m signifies the polynomials order.

- $[\beta_1, \beta_2, \ldots, \beta_m, a_1, a_2, \ldots, a_n, b_0, b_1, \ldots, b_m, c_1, c_2, \ldots, c_l, d_1, d_2, \ldots, d_l]$ are the parameters to be estimated.

The process of identifying the Hammerstein model encompasses of four steps:

1. Testing of the Design

   To ensure the identifiability of the parameters, an identification test needs to be formulated using appropriate excitation signals. Ideally, the test should be tailored to yield an identified model well-suited for control applications. Let the data that has been obtained as input/output be represented as:

   $$\{u(1), y(1), u(2), y(2), \ldots, u(N), y(N)\}$$

   where N denotes the sample number.

2. Estimating the Parameters Creating the model parameters involves utilizing input/output data collected during testing, accomplished through the minimization of the loss function:

   $$V_{PE} = \sum_{t=1}^{N} \varepsilon^2(t) \tag{11}$$

   where

   $$\begin{aligned} \varepsilon(t) &= H^{-1}(q)[y(t) - G(q)f(u(t))] \\ &= \frac{D(q)}{C(q)}[y(t) - \frac{B(q)}{A(q)} \sum_{k=1}^{m} \beta_k u^k(t)] \end{aligned} \tag{12}$$

3. Selecting the order: The values for l and m should be selected to achieve the highest accuracy for control purposes in the obtained model.

4. Validating of the model: The identified model undergoes verification to assess its suitability for control; if not deemed suitable, another identification test is designed. The linear component upper error bound is utilized for this evaluation.

We determine the Hammerstein model by using actual data from the system's experimental tests. The data is entered into the system identification toolbox in MATLAB. The investigations involve regulating the air and water velocities to fall within predetermined ranges. The gas exhibits a velocity range of 0 to 18.288 m/s (0-60 ft/s), while the liquid demonstrates a velocity range of 0 to 3.048 m/s (0-10 ft/s). We specify the input as the ratio of the velocities of the liquid and gas. Input signals and a comparison of pressure values between measured and projected data are illustrated in Figs. 15 and 16. The parameters of the linear model are denoted by the equations that follow, in which F denotes the nonlinear function that involves the unit delay operator q^{-1}. A satisfactory fitting $R = 92.07\%$ is attained between the measured and predicted signals. However, simulation errors lead to results that are noisy when there are sudden changes, in contrast to the actual measured values. The nonlinearity parameters are determined by employing a 1×1 array of Sigmoid Network consisting of 200 units in MATLAB, which is then transformed into a polynomial of order 3. Equation 14 delineates the linear component of the Hammerstein model. We set the parameters for output nonlinearity with a unit gain, in line with the Hammerstein model's requirements. The temperature is considered a disturbance in the system.

$$w(t) = f(u(t)) \tag{13}$$

and the output signal is given as:

$$\begin{aligned} y(t) &= \frac{B(z)}{F(z)} w(t) + e(t) \\ where,\ B(z) &= -0.0943 z^{-1} + z^{-2} \\ F(z) &= 1 - 1.8996 z^{-1} + 1.3422 z^{-2} - 0.4371 z^{-3} \end{aligned} \tag{14}$$

$\frac{B(z)}{F(z)}$ is the transfer function that represents the linear component of the Hammerstein model, while $w(t)$ represents the nonlinear function.

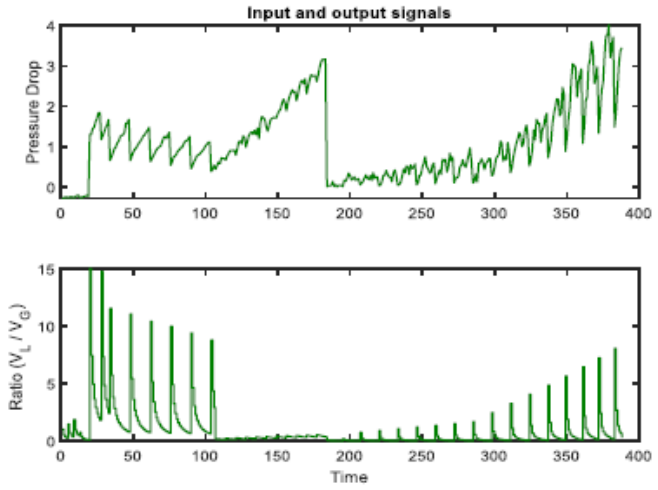

Fig. 15. Output (the upper figure) and input (the lower figure) signals

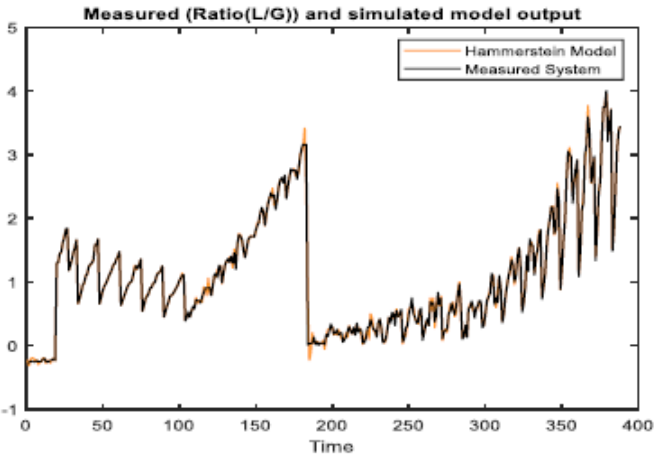

Fig. 16. Measured (pressure drop) and simulated model output

3.4. Summary

According to [34], the Beggs-Brill correlation [4] was found to overpredict the two-phase pressure drop in most air-water datasets. However, the error margin decreased with larger pipe diameters and in the stratified flow regime. For instance, the high-pressure data observed by Andrews [35], as shown in Fig.17, fell within a $\pm 20\%$ error margin.

Table 3. Summary of Identification Techniques

Identification Technique	System	Structure	% Fitting
Neural Network-Based Model	MISO	3 hidden layers with ReLU activation function	99.4%
Nonlinear ARX model	MISO	Order 2 polynomial + SVM	78.77%
	MISO	Order 2 polynomial + 25 units sigmoid network	86.52%
Nonlinear Hammerstein model	MISO	Order 4 transfer function + 10 units sigmoid network	66.9%
	MISO	Order 3 transfer function + 15 units sigmoid network	80.12%
Nonlinear Hammerstein SISO model	SISO	Order 3 transfer function + 200 units sigmoid network	92.07%

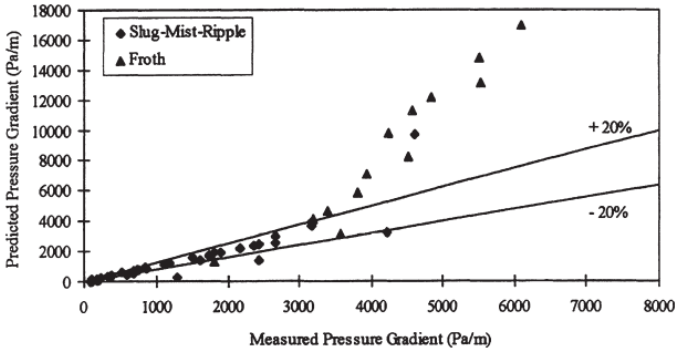

Fig. 17. Comparison of measured and predicted pressure drop for Andrews' 0.0525 m i.d. data [35] using the Beggs–Brill correlations [4]

This section summarizes different identification techniques employed for system modeling, showcasing their associated structures and corresponding fitting percentages. The neural network-based model for multi-input single-out (MISO) systems, featuring three hidden layers with a Rectified Linear Unit (ReLU) activation function, exhibits a high fitting percentage of 99.4%. In the nonlinear ARX category for MISO systems, two distinct structures are presented: one utilizing an Order 2 polynomial combined with Support Vector Machines (SVM) using Gaussian kernel achieving a fitting of 78.77%, and another employing an Order 2 polynomial with a 25-unit sigmoid network yielding an 86.52% fitting. The nonlinear Hammerstein models for MISO systems involve various structures, including order 4 transfer function with a 10-unit sigmoid network (66.9% fitting) and order 3 transfer function with a 15-unit sigmoid network (80.12% fitting). Lastly, a single input single output (SISO) non-linear Hammerstein model is specified, featuring an Order 3 transfer function with a 200-unit sigmoid network and achieving a fitting percentage of 92 07%. These models are designed for system identification, with fitting percentages serving as indicators of their effectiveness in capturing the underlying dynamics of the respective systems. The selection of a particular model and its structure is influenced by the unique characteristics and requirements of the system under consideration.

4. Conclusion

This study offers a comparative analysis of data-driven approaches—specifically, Neural Networks, nonlinear ARX, and Hammerstein-Wiener models—for predicting pressure drops in multi-phase flow systems. The authors conducted extensive experiments to collect data that informed the development and comparison of these models. While Neural Networks (NNs) are effective at capturing complex nonlinearities, they are highly susceptible to overfitting, especially in scenarios with limited or highly variable data, as is often the case in multi-phase flow systems. To mitigate this risk, the study focuses on non-linear ARX and Hammerstein-Wiener models, which demonstrated robust predictive capabilities and greater generalizability, providing a more reliable solution for prediction of pressure drops. By leveraging experimental data in data-driven modeling, our approach effectively addresses the inherent nonlinearities and variability in multi-phase flow systems, enhancing predictive capabilities and offering a robust solution for dynamic pressure management. The findings of this research not only demonstrate the potential of advanced machine learning models in industrial applications, but also underscore the importance of model selection tailored to specific system characteristics. Future studies should focus on expanding these models with larger and more diverse datasets and applying them to real-time flow control systems. The insights gained here set a foundation for the development of next-generation control algorithms, ultimately contributing to safer, more efficient multi-phase flow system operations.

Data Availability Statement

The text includes all the necessary references to the data that was used.

Acknowledgments

The authors acknowledge the support of the Deanship of Scientific Research, King Fahd University of Petroleum and Minerals. The authors also express their sincere appreciation to Dr. Mostafa Alnaser from Yokogawa, Saudi Arabia, and Dr. Rahul N. Gajbhiye from Collage of Petroleum Engineering and Geoscience (CPG), KFUPM, for generously sharing the experimental data.

Conflicts of Interest

The authors declare no conflicts of interest.

References

1. Al-Dogail, Ala S., and Rahul N. Gajbhiye. "Effects of density, viscosity and surface tension on flow regimes and pressure drop of two-phase flow in horizontal pipes." Journal of Petroleum Science and Engineering 205 (2021): 108719.
2. AL-Dogail, Ala, et al. "Dimensionless artificial intelligence-based model for multiphase flow pattern recognition in horizontal pipe." SPE Production & Operations 37.02 (2022): 244-262.
3. Aziz, Khalid, and George W. Govier. "Pressure drop in wells producing oil and gas." Journal of Canadian Petroleum Technology 11.03 (1972).
4. Beggs, Dale H., and James P. Brill. "A study of two-phase flow in inclined pipes." Journal of Petroleum technology 25.05 (1973): 607-617.
5. Mukherjee, Hemanta, and James P. Brill. "Liquid holdup correlations for inclined two-phase flow." Journal of Petroleum Technology 35.05 (1983): 1003-1008.
6. Simpkins, Alex. "System identification: Theory for the user, (ljung, l.; 1999)[on the shelf]." IEEE Robotics & Automation Magazine 19.2 (2012): 95-96.
7. Rukandi M. F., Pratama, S. H., Suprijanto, and Muchtadi, F. I., "Movement imagery process in brain-computer interface based on mu rhythm," in 2nd Int. Conf. Instrum. Control Autom., 2009, pp. 294–297.
8. ElFerik, Sami, et al. "OKID-based identification and control of unmanned ducted-fan helicopter." IFAC Proceedings Volumes 45.1 (2012): 115-120.
9. El Ferik, Sami, et al. "LQG-based control of unmanned helicopter using OKID-based identification approach." 10th International Multi-Conferences on Systems, Signals & Devices 2013 (SSD13). IEEE, 2013.
10. Mulyana, Tatang, et al. "Arx model of four types heat exchanger identification." Proceedings of MUiCET. 2011.
11. Sadikin, Azmahani, Muhammad Adib Jamil, and Norasikin Mat Isa. "Investigation of Boiling Heat Transfer and Flow Regimes in a Heat Exchanger." Applied Mechanics and Materials 660 (2014): 679-683.
12. Al-Naser, Mustafa, Moustafa Elshafei, and Abdelsalam Al-Sarkhi. "Artificial neural network application for multiphase flow patterns detection: A new approach." Journal of Petroleum Science and Engineering 145 (2016): 548-564.
13. Najafi, Behzad, et al. "Machine learning based models for pressure drop estimation of two-phase adiabatic air-water flow in micro-finned tubes: Determination of the most promising dimensionless feature set." Chemical Engineering Research and Design 167 (2021): 252-267.
14. AL-Dogail, Ala Shafeq, et al. "Machine learning approach to predict pressure drop of multi-phase flow in horizontal pipe and influence of fluid properties." SPE Gas & Oil Technology Showcase and Conference. SPE, 2023.
15. Ebrahimi, A., and E. Khamehchi. "A robust model for computing pressure drop in vertical multiphase flow." Journal of Natural Gas Science and Engineering 26 (2015): 1306-1316.
16. Garcia, Juan Jose, et al. "Prediction of pressure drop during evaporation of R407C in horizontal tubes using artificial neural networks." International Journal of Refrigeration 85 (2018): 292-302.
17. Ahmadi, Mohammad Hossein, et al. "Prediction of the pressure drop for CuO/(Ethylene glycol-water) nanofluid flows in the car radiator by means of Artificial Neural Networks analysis integrated with genetic algorithm." Physica A: Statistical mechanics and its Applications 546 (2020): 124008.
18. Barroso-Maldonado, J. M., et al. "ANN-based correlation for frictional pressure drop of non-azeotropic mixtures during cryogenic forced boiling." Applied Thermal Engineering 149 (2019): 492-501.
19. Bawazir, Aiman F., et al. "Adaptive PID Controller Using Neural Network for Pressure Drop in Nonlinear Fluid Systems." 2024 21st International Multi-Conference on Systems, Signals & Devices (SSD). IEEE, 2024.
20. Shanthi, C., and N. Pappa. "An artificial intelligence based improved classification of two-phase flow patterns with feature extracted from acquired images." ISA transactions 68 (2017): 425-432.
21. Xiao, Jian, et al. "Using artificial intelligence to improve identification of nanofluid gas–liquid two-phase flow pattern in mini-channel." AIP Advances 8.1 (2018).
22. Saptomo, Satyanto Krido, et al. "Experimental and numerical investigation of laboratory scale sheetpipe-typed automatic subsurface irrigation." Communications in Science and Technology 6.2 (2021): 117-124.
23. Kanin, E. A., et al. "A predictive model for steady-state multiphase pipe flow: Machine learning on lab data." Journal of Petroleum Science and Engineering 180 (2019): 727-746.
24. Söderström, Torsten. "Identification of stochastic linear systems in presence of input noise." Automatica 17.5 (1981): 713-725.
25. Zhang, Jinxi, et al. "Identification of errors-in-variables ARX model with time varying time delay." Journal of Process Control 115 (2022): 134-144.
26. Gurney, Kevin. "An introduction to neural networks." CRC press, 2018.
27. Abiodun, Oludare Isaac, et al. "State-of-the-art in artificial neural network applications: A survey." Heliyon 4.11 (2018).
28. Shanshiashvili, B., A. Prangishvili, and Z. Tsveraidze. "Structure Identification of Continuous-Time Block-Oriented Nonlinear Systems in the Frequency Domain." IFAC-PapersOnLine 52.13 (2019): 463-468.
29. Ohata, Akira, Katsuhisa Furuta, and Hiroaki Nita. "Identification of nonlinear ARX model with input and output dependent coefficients." 2006 IEEE Conference on Computer Aided Control System Design, 2006 IEEE International Conference on Control Applications, 2006 IEEE International Symposium on Intelligent Control. IEEE, 2006.
30. Sun, Lijie, et al. "A robust hammerstein-wiener model identification method for highly nonlinear systems." Processes 10.12 (2022): 2664.
31. Dai, Yingli, Dequan Li, and Dong Wang. "Review on the nonlinear modeling of hysteresis in piezoelectric ceramic actuators." Actuators. Vol. 12. No. 12. MDPI, 2023.
32. Schoukens, Maarten, and Koen Tiels. "Identification of block-oriented nonlinear systems starting from linear approximations: A survey." Automatica 85 (2017): 272-292.
33. Amin, Al, et al. "Nonlinear model predictive control of a Hammerstein Weiner model based experimental managed pressure drilling setup." ISA transactions 88 (2019): 225-232.
34. Spedding, P. L., Emmanuel Bénard, and G. F. Donnelly. "Prediction of pressure drop in multiphase horizontal pipe flow." International communications in heat and mass transfer 33.9 (2006): 1053-1062.
35. Andrews, Donald Edgar, et al. "The Prediction of Pressure Loss During Two-Phase Horizontal Flow in Two-Inch Pipe." (1967): 44-51.