Journal of
Marine Science and Engineering

MDPI

Article

Modeling Fluid Flow in Ship Systems for Controller Tuning Using an Artificial Neural Network

Nur Assani [1,*], Petar Matić [1], Danko Kezić [1] and Nikolina Pleić [2,3]

[1] Department for Marine Electrical Engineering and Information Technologies, Faculty of Maritime Studies, University of Split, Ruđera Boškovića 37, 21000 Split, Croatia; pmatic@pfst.hr (P.M.); dkezic@pfst.hr (D.K.)
[2] Department of Medical Biology, School of Medicine, University of Split, Šoltanska 2, 21000 Split, Croatia; npleic@mefst.hr
[3] Department of Computer Engineering, Aspira University of Applied Sciences, Mike Tripala 6, 21000 Split, Croatia
[*] Correspondence: nassani@pfst.hr; Tel.: +385-(0)97-694-2401

Abstract: Flow processes onboard ships are common in order to transport fluids like oil, gas, and water. These processes are controlled by PID controllers, acting on the regulation valves as actuators. In case of a malfunction or refitting, a PID controller needs to be re-adjusted for the optimal control of the process. To avoid experimenting on operational real systems, models are convenient alternatives. When real-time information is needed, digital twin (DT) concepts become highly valuable. The aim of this paper is to analyze and determine the optimal NARX model architecture in order to achieve a higher-accuracy model of a ship's flow process. An artificial neural network (ANN) was used to model the process in MATLAB. The experiments were performed using a multi-start approach to prevent overtraining. To prove the thesis, statistical analysis of the experimental results was performed. Models were evaluated for generalization using mean squared error (MSE), best fit, and goodness of fit (GoF) measures on two independent datasets. The results indicate the correlation between the number of input delays and the performance of the model. A permuted k-fold cross-validation analysis was used to determine the optimal number of voltage and flow delays, thus defining the number of model inputs. Permutations of training, test, and validation datasets were applied to examine bias due to the data arrangement during training.

Keywords: artificial neural network; NARX model; ship's fluid flow process; controller tuning; data-based modelling; digital twin

check for updates

Citation: Assani, N.; Matić, P.; Kezić, D.; Pleić, N. Modeling Fluid Flow in Ship Systems for Controller Tuning Using an Artificial Neural Network. *J. Mar. Sci. Eng.* **2024**, *12*, 1318. https://doi.org/10.3390/jmse12081318

Academic Editor: Burak Can Cerik

Received: 10 July 2024
Revised: 31 July 2024
Accepted: 2 August 2024
Published: 4 August 2024

1. Introduction

Digital twins (DTs) rely on measurements to update the model of the system obtained by identification, as described in [1]. Although this model does not provide insight into the system's physical properties, it is easier to develop. It also better describes the system's input–output behavior, making it suitable for control system design as noted in [2].

The quality of controller adjustment in that case depends on the quality of the model. However, in [3], the model developed through identification, despite its high accuracy, does not yield precise controller tuning. This suggests that other identification methods should be tested to produce a model with a higher level of accuracy.

Artificial neural networks (ANNs) are a flexible, data-driven tool that can be used for different applications across different fields of science, if adequate data are available, as stated in [4], a paper that presents the analyses of the applications of artificial neural networks on ships.

In [5], an ANN Nonlinear Autoregressive Exogenous (NARX) model was used as a reference model for an adaptive control scheme to improve disturbance rejection of a PID controller. Similarly, an ANN NARX model was used for online PID controller tuning in [6].

A similar model was used in [7] to predict the time series of the mooring line tension of a floating production storage and offloading platform at sea. Ultimately, PID controller tuning was performed in [8] using the NARX model of a steam distillation plant.

This paper presents the first part of a modeling process, input determination, that aims to produce a highly accurate ANN NARX model of a flow process that would enable an accurate DT-based controller tuning for a ship's flow processes. As described in [9], when using an ANN to form a model, it is recommended to use a systematic approach that includes goal definition, determination of input and output variables, selection of a network type, number of neurons (in each layer), training algorithm, data division, and data processing, as well as the evaluation measure.

During training, data have been divided into train, test, and validation subsets to prevent overtraining, i.e., to improve the generalization abilities of the model, as described in [10]. Six different arrangements of training, test, and validation datasets have been examined to make relevant statistical comparisons of the results and determine the influence of data division on model performances. Two additional tests were performed with data that had not been presented to the network during training to examine its generalization abilities.

Evaluation measures included the mean squared error (MSE) as a cost function during training as well as the measure of evaluating the network performance on additional sets of data. The goodness of fit (GoF) measure was also used to evaluate the networks, as in [2], to make the results comparable with previous work. The best fit was used as an additional evaluation measure, as in [11], where the ANN NARX model of a turboshaft turbine engine fuel control unit was developed.

The best-evaluated network was chosen as a representative model based on a statistical analysis of the evaluation results, providing insights into the optimal number of input delays (n and m) for system identification purposes. The chosen model was further tested on two additional datasets, previously used to evaluate the model developed in [2], to demonstrate that the proposed modeling method in this paper achieves a higher level of accuracy.

The remainder of the paper is organized as follows: Section 2 describes the materials and methods used in this paper, including the description of the flow control system, NARX model, model training, and cross-validation process. Section 3 presents the model evaluation results and compares the best-performing model to a previously developed model. Section 4 highlights the insights provided by the results presented in this paper. Finally, Section 5 concludes the paper and highlights future work.

2. Materials and Methods

2.1. Flow Control System

The flow control system consists of a main water tank (1), single-phase electric motor-driven centrifugal pump (2), flow meter (3), solenoid valve (4), flow controller (5), disturbance ball valve (6), secondary water tank (7), drain ball valve (8), and corresponding copper piping, as shown in Figure 1.

Water is pumped from the main water tank (1) by a single-phase centrifugal pump (2). Water flow is measured by a flow meter (3) and fed into the PID flow controller (5), which operates the solenoid valve (4) according to the difference between the set-point and the measured flow. Water is pumped through copper pipes and the disturbance ball valve (6) into the secondary water tank (7) and back to the main water tank through the drain ball valve (8).

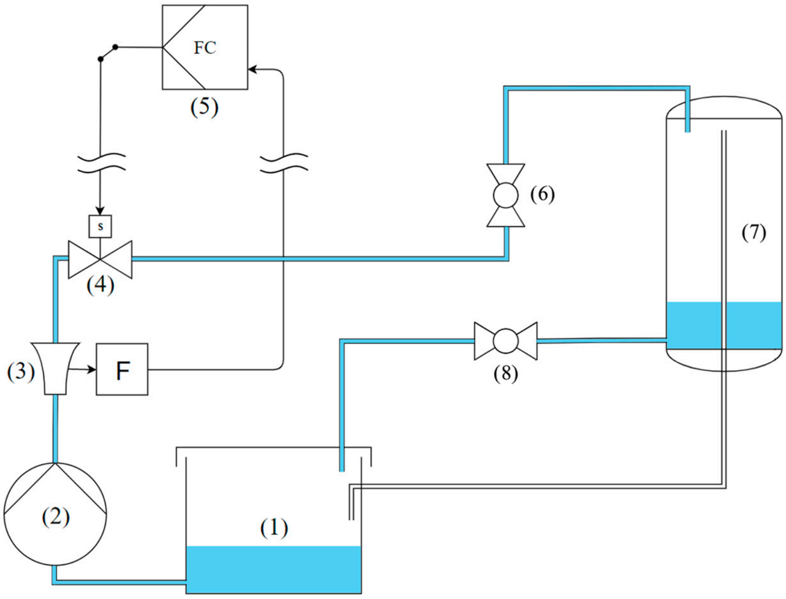

Figure 1. Flow control system.

2.2. NARX Model

The NARX model is commonly used for nonlinear dynamic system approximation purposes, as stated in [12]. The mathematical representation of an NARX model is defined by the expression (1), where $u(t) \in \mathrm{N}$ and $y(t) \in \mathrm{N}$ represent discrete input and output variables of the model at time step t, respectively. The number of input and output delays, i.e., the number of input and output samples, are presented by n and m, respectively, where $n \geq 0$ and $m \geq 1$.

$$y\ (t+1) = f[y(t), \ldots, y(t-m+1)\ ;\ u(t),\ u(t-1), \ldots, u(t-n+1)] \tag{1}$$

Function f can be modeled by a feed-forward artificial neural network using one hidden layer of neurons with sigmoid activation function, as it can approximate any continuous function with good accuracy, according to [13].

2.3. Data Acquisition and Preparation

The data have been recorded with the sampling rate of two samples per second using the data acquisition interface module, as shown in Figure 2, and divided into three independent datasets. The first dataset consists of 1514 samples, while the second and third consist of 598 and 266 samples. Therefore, the same datasets used in previous work [2] are used in this paper to conduct valid and comparable analyses of the results.

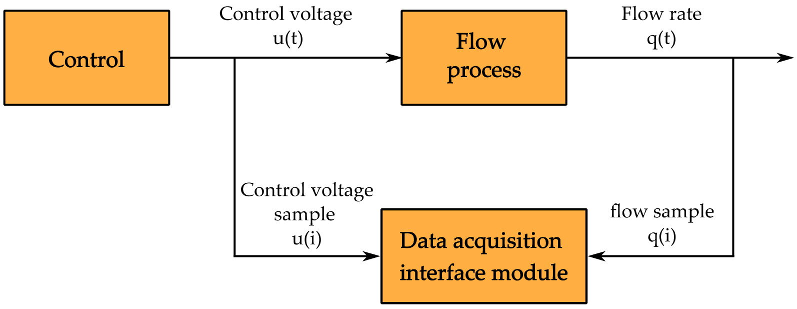

Figure 2. Flowchart of the data sampling.

2.4. ANN NARX Model Training

The NARX model is formed using a feed-forward ANN, a multilayer perceptron, in MATLAB, while the training was performed using the Bayesian Regularization (BR) algorithm with a starting number of 100 neurons in a hidden layer. As an optimizer during the training process, backpropagation has been used. The model uses voltage ($u_{\mathrm{t-n}}$) and flow ($q_{\mathrm{t-m}}$) time series as input variables and outputs the current flow (q_{t}), as defined by Equation (2).

$$q_t = \mathrm{ANN}(u_{t-n}, q_{t-m}) \tag{2}$$

Multiple training iterations of the model were performed to determine the optimal number of inputs (n and m) required to achieve high model accuracy, as illustrated in Figure 3. To avoid a local minimum of cost function during training, a multi-start was applied, meaning that each network structure was trained 10 times, based on a recommendation from [14]. The number of epochs for each training attempt was set to 10.000. A total of 1514 time-series data samples have been divided into training, test, and validation datasets, containing 70%, 15%, and 15% of data samples, respectively.

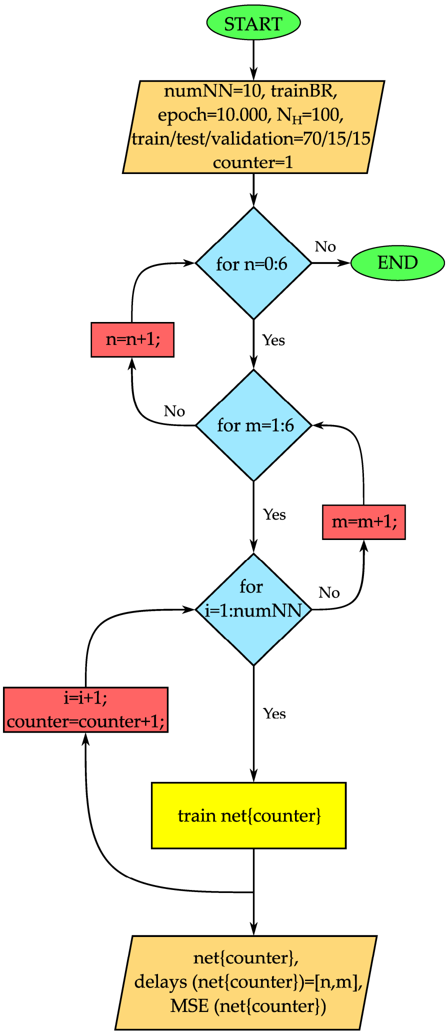

Figure 3. Flowchart of the proposed methodology for training the ANN NARX models.

As a result of one training iteration based on the flowchart shown in Figure 3, 420 ANN NARX models were stored in a cell *net{}* along with 420 combinations of the delays *m* and *n*, stored in a cell *delays(net{})*. For each trained model, MSE was calculated during the training process and stored in cell *MSE(net{})*.

The training was performed using a desktop computer based on the Windows 10 operating system with 32 GB DDR4 3200 memory, an Intel i7-9700K processor, and 500 GB of NVMe M2 SSD storage. The required time to develop all 2520 models was 38 h and 56 min.

2.5. Permuted k-Fold Cross-Validation with Train–Validation–Test Splits

As is stated in [15], when working on smaller datasets such as in this paper, k-fold cross-validation is an ideal choice for model evaluation. Such a validation method implies partitioning data into k segments or folds, each used to train the ANN NARX model. After repeating the training process for all k folds, k performance estimates are obtained for each evaluation measure, i.e., k performance estimates for MSE, k performance estimates for the GoF, and k performance estimates for best-fit evaluation measure. These estimates are further averaged to provide a single performance measure for each evaluation measure, thus providing a more reliable indicator of the model's effectiveness compared to a single fold of input data.

As there is no formal rule for selecting the number of k folds, it is usually chosen to be between 5 and 10, as stated in [16]. Considering the aforementioned, and since there are six possible permutations of input data, a permuted 6-fold cross-validation was performed to evaluate the performance of the trained ANN NARX models. Such a k number should offer a good balance between bias and variance.

Therefore, to avoid biased evaluation estimates in the testing and validation processes, the networks were trained using all possible permutations of the training, testing, and validation datasets across the entire dataset, keeping a 70/15/15 ratio, as shown on the flowchart in Figure 4.

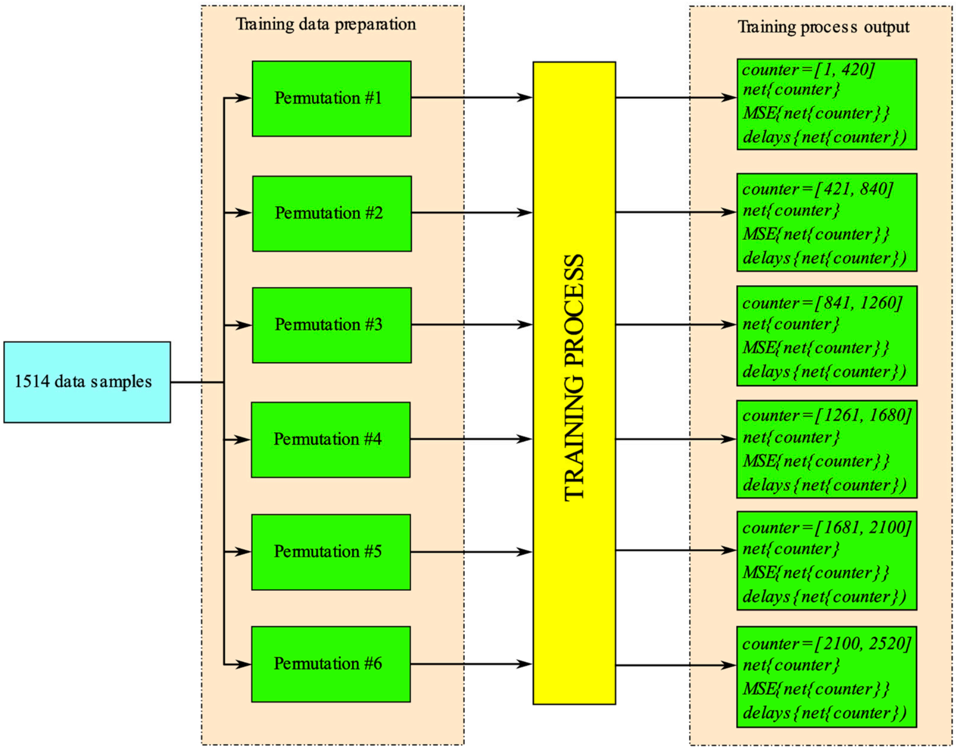

Figure 4. Flowchart of the data preparation methodology and training of the ANN NARX models.

For each training iteration, 70% of the 1514 data samples have been used for the training dataset. The remaining 30% of the data samples were used for the test and validation datasets, each consisting of 15% of the 1514 data samples. Therefore, there were six possible permutations without repetition of the datasets, as shown in Table 1.

Given that each training iteration produced 420 ANN NARX models, and there were six permutations of the datasets, 2520 ANN NARX models were developed.

Table 1. Permutations of training, test, and validation datasets within the overall dataset containing 1514 samples.

Permutation Number	Training Dataset [Sample Sequence]	Test Dataset [Sample Sequence]	Validation Dataset [Sample Sequence]
1.	[1, 1060]	[1061, 1287]	[1288, 1514]
2.	[1, 1060]	[1288, 1514]	[1061, 1287]
3.	[228, 1287]	[1, 227]	[1288, 1514]
4.	[228, 1287]	[1288, 1514]	[1, 227]
5.	[455, 1514]	[1, 227]	[228, 454]
6.	[455, 1514]	[228, 454]	[1, 227]

To confirm the performance of the trained networks, each network was given two new time-series datasets containing 598 and 266 data samples. The process of testing is shown in the flowchart in Figure 5.

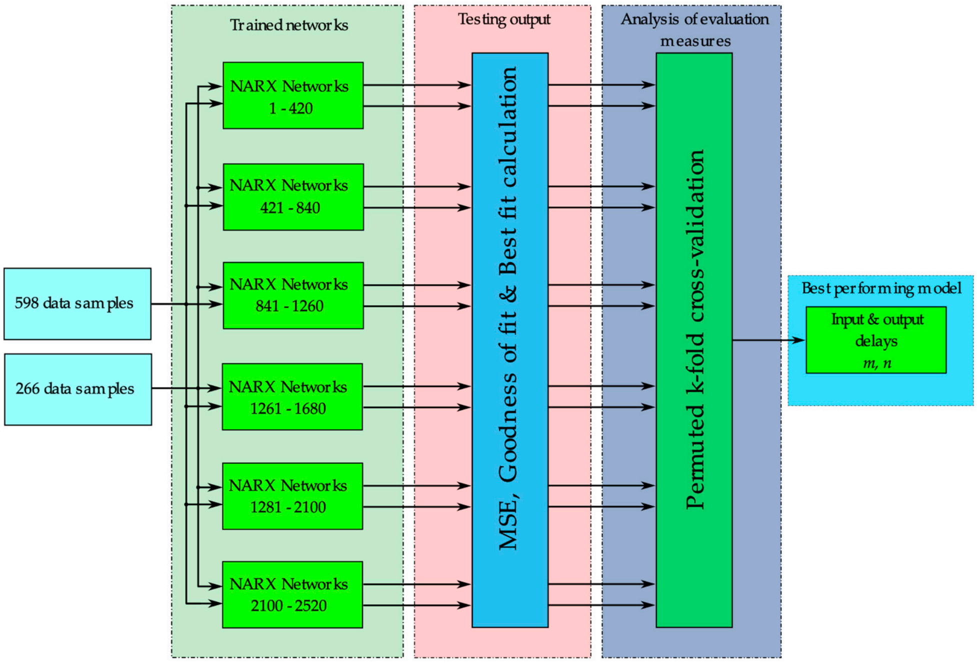
Trained networks
Testing output
Analysis of evaluation measures
NARX Networks 1 - 420
NARX Networks 421 - 840
NARX Networks 841 - 1260
NARX Networks 1261 - 1680
NARX Networks 1281 - 2100
NARX Networks 2100 - 2520
598 data samples
266 data samples
MSE, Goodness of fit & Best fit calculation
Permuted k-fold cross-validation
Best performing model
Input & output delays
m, n

Figure 5. Flowchart of the ANN NARX models' testingon two new time–series datasets.

The MSE was used to measure the amount of error of the ANN NARX model during training, as well as in the validation and testing processes following Equation (3), where s represents the number of samples in the dataset, q_i represents the measured flow output for each input sample in the dataset, and $\hat{q}_i$ represents the ANN NARX model prediction of flow for given input sample i. GoF and best fit were also used according to the Equations (4) and (5), using the same variable annotations as in the MSE equation with $\overline{q}$ representing the mean of the measured flow in the dataset.

$$MSE = \frac{1}{s}\sum_{i=1}^{s}\left(q_{\,i} - \hat{q}_i\right)^2 \tag{3}$$

$$Goodness\ of\ fit = \left(1 - \sqrt{\frac{\sum_{i=1}^{s}(q_i - \hat{q}_i)^2}{\sum_{i=1}^{s}(q_i - \overline{q})^2}}\right) \times 100\% \tag{4}$$

$$Best\ fit = \left(1 - \frac{\sum_{i=1}^{s}|q_i - \hat{q}_i|}{\sum_{i=1}^{s}|q_i - \overline{q}|}\right) \times 100\% \tag{5}$$

3. Results

As mentioned before, the MSE, GoF, and best fit serve as crucial metrics in this paper. However, in order to determine the best-performing model, only one measure can be used for ranking. The MSE was used for the performance evaluation of the trained models, as it served as a cost function during the training process.

3.1. Model Ranking Based on the MSE Evaluation Measure

The ten best-performing ANN NARX models on two independent datasets consisting of 598 and 266 data samples are presented in Tables 2 and 3. The ranking was based on the MSE, while the two remaining evaluation measures are also presented in the tables. The ANN NARX number column shows at which training iteration and permutation of input data the best-performing network occurred. As can be seen in Tables 2 and 3, the ANN NARX models with 6 flow delays and two or five voltage delays are in the top three solutions for both datasets. To select the best-performing model overall, an average MSE has been calculated for those two models and presented in Table 4.

Table 2. Best-performing ANN NARX models using test dataset 1 (598 samples).

Flow Delay (m)	Voltage Delay (n)	ANN NARX Number (Permutation)	MSE	Best Fit	GoF
6	1	366 (1st)	77,214	98,195	96,744
6	5	402 (2nd)	77,659	98,128	96,724
6	2	380 (2nd)	82,881	98,208	96,573
6	5	405 (2nd)	87,583	98,018	96,551
5	3	322 (2nd)	88,515	98,092	96,534
5	3	329 (2nd)	91,625	98,658	96,376
6	1	367 (1st)	92,666	98,569	96,355
5	2	313 (1st)	95,070	98,576	96,304
6	1	365 (2nd)	95,491	98,233	96,300
6	1	361 (2nd)	96,286	98,744	96,296

Table 3. Best-performing ANN NARX models using test dataset 2 (266 samples).

Flow Delay (m)	Voltage Delay (n)	ANN NARX Number (Permutation)	MSE	Best Fit	GoF
6	2	371 (1st)	53,460	98,059	96,885
6	5	403 (2nd)	62,642	97,900	96,506
5	4	339 (2nd)	63,038	97,584	96,519
5	4	336 (2nd)	63,522	97,756	96,631
5	3	329 (1st)	66,465	97,744	96,502
6	5	408 (2nd)	66,664	98,038	96,575
5	2	315 (1st)	67,343	97,636	96,474
6	5	405 (2nd)	69,844	97,456	96,316
4	4	278 (1st)	70,027	97,980	96,943
6	5	402 (3rd)	70,335	97,794	96,295

Table 4. The best-performing ANN NARX models in both datasets.

Flow Delay (m)	Voltage Delay (n)	Average MSE
6	2	68,171
6	5	70,151

Convergence of the MSE during the training process for the network configuration with six flow delays and two voltage delays is shown in Figure 6.

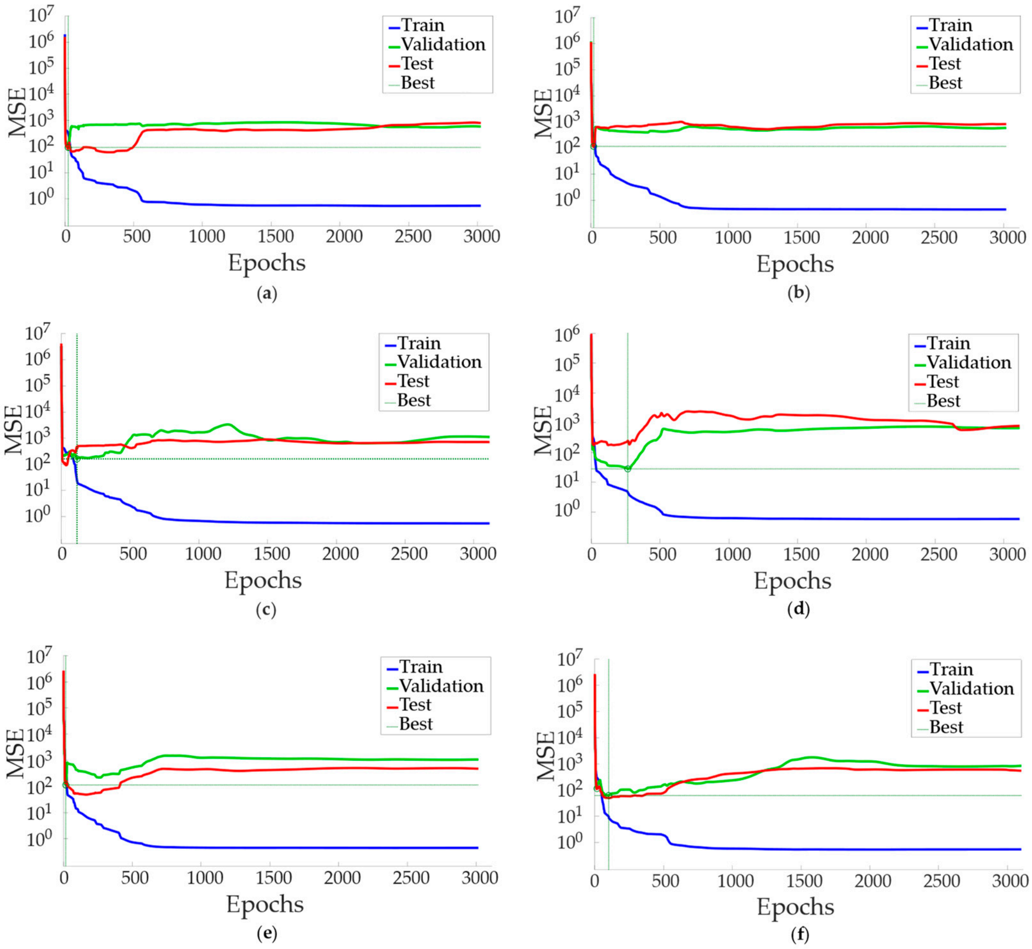

Figure 6. Convergence of the MSE during training: (**a**) using 1st permutation of data; (**b**) using 2nd permutation of data; (**c**) using 3rd permutation of data; (**d**) using 4th permutation of data; (**e**) using 5th permutation of data; and (**f**) using 6th permutation of data.

3.2. Comparison of the ANN NARX Model and Transfer Function Model

The best-performing ANN NARX model with six flow and two voltage delays is selected based on the results presented in this paper and is compared to the previously developed transfer function (TF) model presented in [2]. Models were tested on two datasets using the GoF measure as it was the only measure used in previous research. Comparison results are presented in Table 5, and the time-response graphs are shown in Figures 7 and 8. The results of the statistical analysis indicate that the newly presented ANN NARX model significantly outperforms the previous TF model, with the paired-sample t-test showing a p-value of 0.016 (t = 39.839, df = 1)

Error analysis has been performed on both test datasets to identify where the model underperforms compared to the target values. Since the results are similar for both datasets, error analysis on test dataset 2 has been presented in this paper. As can be noted in Figure 9, the ANN NARX model makes errors during the transition process, while making minimal errors at the steady state.

Table 5. Best-performing ANN NARX model compared to previously developed transfer function model.

Dataset	ANN NARX Model GoF	TF Model GoF
Test dataset #1	97.85%	91.83%
Test dataset #2	97.53%	91.20%

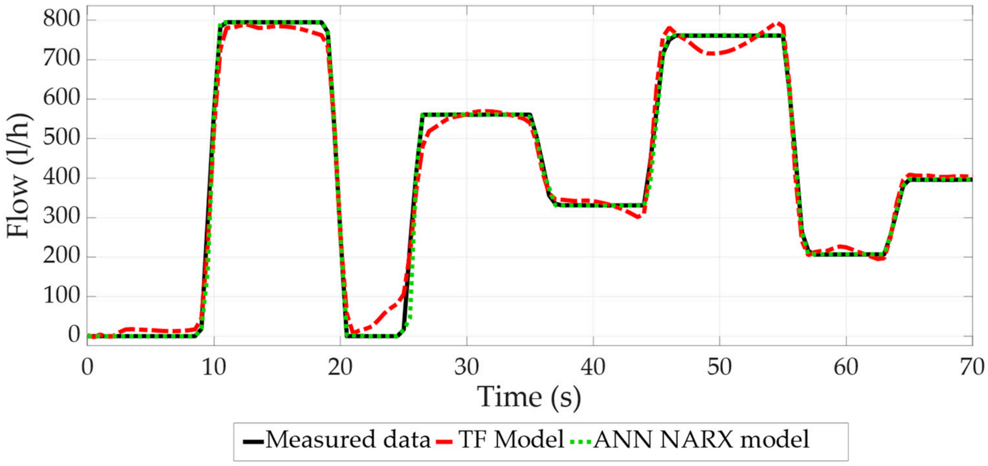

Figure 7. Time-response of the best-performing ANN NARX model compared to the previously developed TF model using additional test dataset 1.

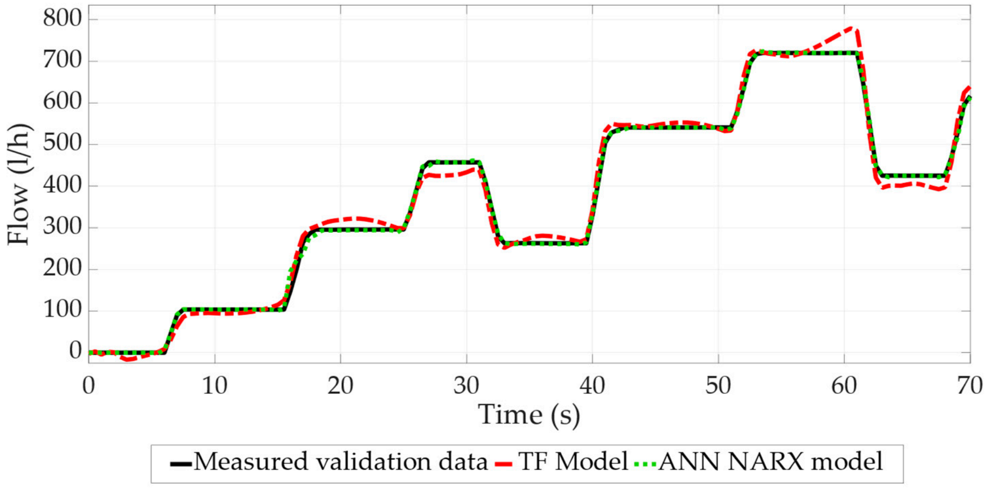

Figure 8. Response time of the best-performing ANN NARX model compared to the previously developed TF model using additional test dataset 2.

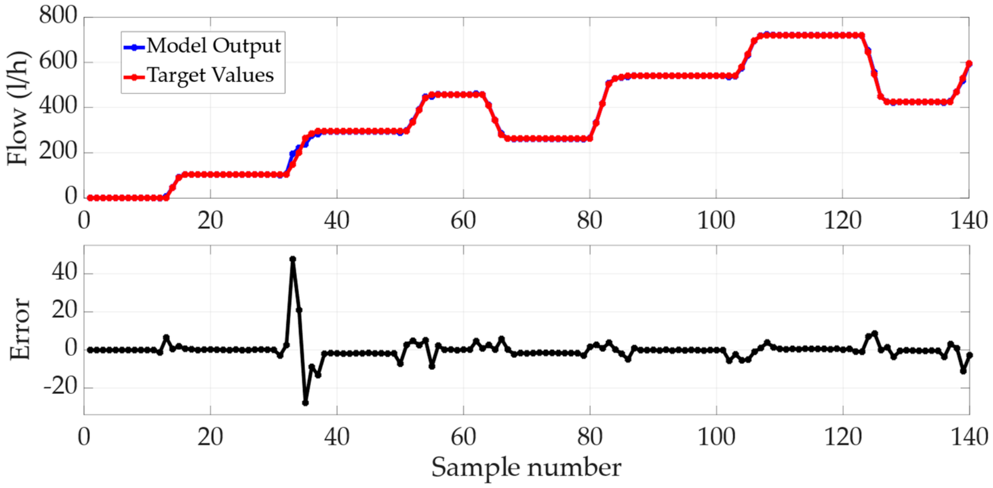

Figure 9. Error plot of the best-performing ANN NARX model compared to the target values from test dataset 2.

4. Discussion

The main purpose of this paper was to develop and evaluate an ANN NARX model of a real system for controller tuning purposes. The motive for this paper was the limitation noted in the previously developed transfer function model, which, despite achieving a high GoF (91.2%), underperformed in controller tuning applications.

The experiments were conducted to determine the optimal number of input delays (n and m), i.e., the number of inputs resulting in the creation of 2520 ANN NARX models in total. The number of experiments, i.e., the models produced, were influenced by the idea of increasing the generalization abilities and robustness of the model.

The results presented in this paper provide several important insights:

- Performance Across Models: The ANN NARX models exhibited a wide range of performance levels, indicating that the choice of input delays significantly impacts the model's ability to accurately capture the system dynamics.
- Model Selection: Some configurations of input delays (n, m pairs) led to highly accurate models, while others did not enable such a good performance. The evaluation of the first test dataset highlighted several models with high performance. Notably, the model with six flow delays and one voltage delay (ANN NARX 366 using the first permutation) achieved the lowest MSE of 77.21, followed by the model with six flow and five voltage delays and the model with six flow and two voltage delays. This suggests a lower sensitivity of the model output to the voltage time series, and a higher importance of its latter value than its trend. On the other hand, the flow shows quite the opposite characteristic, emphasizing the importance of the flow time series. The second test dataset further confirmed the ability of the ANN NARX models to capture the nonlinear dynamics of the system and also helped in determining the optimal number of inputs.
- Model Robustness: The extensive search over the (n, m) space and multiple data permutations provided a robust framework for identifying the optimal ANN NARX model configuration and examined sensitivity to different arrangements of the training datasets.
- Comparative Analysis With Transfer Function Model: The comparative analysis between the ANN NARX models and the previously developed transfer function model revealed significant improvements. While the transfer function model achieved a high GoF (91.2%), it failed to provide satisfactory performance in controller tuning applications. In contrast, the ANN NARX models demonstrated consistently high performance across multiple permutations and configurations, with GoF values frequently exceeding 96%.
- Controller Tuning Implications: Higher accuracy and the robustness of the ANN NARX models, demonstrated by their consistent performance across multiple datasets, should be beneficial for controller tuning applications. This finding is important for the development of a high-performance control systems relying on DTs.

5. Conclusions

Through a comprehensive experimentation involving 42 combinations of voltage and flow delays as the input variables, 10 training iterations, and 6 permutations of the input datasets per combination, a total of 2520 ANN NARX flow models were trained and evaluated. Models' performances were assessed using MSE, GoF and best fit measures on two independent datasets. The results demonstrated the high accuracy of the created models. Multiple data permutations ensured that the model was not overfitting to a specific dataset but was capturing the underlying system behavior effectively.

The results indicate that the flow time series, i.e., the flow delay (m), has a strong influence on the ANN NARX model performance, as the first ten best-performing models mainly use five or six previous data samples, thus determining the optimal value of the parameter. Furtherly, the experimental results show that the voltage time series, i.e., the voltage delay (n), does not have such influence on the model performance, as its optimal values range from 1 to 5, thus indicating a higher importance of its more recent values to the output, rather than a longer time series.

On the first independent test dataset, the model with six flow delays and one voltage delay performed the best, but it did not make it to the top ten on the second test dataset, in which the model using six and two delays showed the best performances. Therefore, the best-performing ANN NARX model was determined by averaging the results from two independent tests, and the model using six flow and two voltage delays has been selected.

The different permutations of the input data highlighted the robustness of the ANN NARX models, as the resulting models did not show any significant sensitivity to different permutations of datasets during training.

When comparing the performance of the best-evaluated ANN NARX model to the transfer function model developed in previous work, it is evident that the ANN NARX model generally provided a better fit, while the best-performing ANN NARX model outperforms the best transfer function model by 6% in terms of GoF as a quality measure.

Although the model presented in this paper showed better performances than its transfer function precursor, it still needs to be tested for PID controller tuning, which is the next step planned in further work. In addition, experiments to determine the optimal number of consecutive training iterations and experiments with different ANN structures, training algorithms, and training parameters could be investigated to further improve model performance.

Author Contributions: Conceptualization, P.M. and N.A.; methodology, P.M., N.A. and N.P.; software, N.A.; validation, N.A. and N.P.; formal analysis, N.A.; investigation, N.A.; resources, N.A.; data curation, N.A.; writing—original draft preparation, N.A.; writing—review and editing, P.M.; visualization, N.A.; supervision, P.M.; project administration, D.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The raw data supporting the conclusions of this article will be made available by the authors on request.

Acknowledgments: Research supported by EU Operational Programme Competitiveness and Cohesion through the project "Functional integration of the University of Split, PMF/PFST/KTF through the development of scientific and research infrastructure in the three faculty (3F) building" under contract number: KK.01.1.1.02.0018.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Assani, N.; Matić, P.; Katalinić, M. Ship's Digital Twin—A Review of Modelling Challenges and Applications. *Appl. Sci.* **2022**, *12*, 6039. [CrossRef]
2. Assani, N.; Matic, P.; Kezic, D. Flow Control Process Identification Using Matlab's System Identification Toolbox. In Proceedings of the 2022 8th International Conference on Control, Decision and Information Technologies, CoDIT 2022, Istanbul, Turkey, 17–20 May 2022; Institute of Electrical and Electronics Engineers Inc.: New York, NY, USA, 2022; pp. 1228–1232.
3. Assani, N.; Matić, P.; Kezić, D. Evaluating the Performance of the Well-Known Controller Tuning Methods for the Flow Control Using the Process Model. In Proceedings of the 9th 2023 International Conference on Control, Decision and Information Technologies, CoDIT 2023, Rome, Italy, 3–6 July 2023; Institute of Electrical and Electronics Engineers Inc.: New York, NY, USA, 2023; pp. 433–438.
4. Assani, N.; Matic, P.; Kastelan, N.; Cavka, I.R. A Review of Artificial Neural Networks Applications in Maritime Industry. *IEEE Access* **2023**, *11*, 139823–139848. [CrossRef]
5. Alimohammadi, H.; Alagoz, B.B.; Tepljakov, A.; Vassiljeva, K.; Petlenkov, E. A NARX Model Reference Adaptive Control Scheme: Improved Disturbance Rejection Fractional-Order PID Control of an Experimental Magnetic Levitation System. *Algorithms* **2020**, *13*, 201. [CrossRef]
6. Liu, J.; Li, T.; Zhang, Z.; Chen, J. NARX Prediction-Based Parameters Online Tuning Method of Intelligent PID System. *IEEE Access* **2020**, *8*, 130922–130936. [CrossRef]

7. Nikkhah, E.; Fernandes, A.C.; Caprace, J.-D. Design of Experiments Technique Applied to Artificial Neural Network Models for FPSO Mooring System Analysis. *J. Mar. Sci. Eng.* **2023**, *11*, 2194. [CrossRef]
8. Mohammad, N.N.; Norazman, H.; Hambali, N. NARX Modelling and PID Controller in Monitoring Water Temperature of SDPP. In Proceedings of the 2022 IEEE 13th Control and System Graduate Research Colloquium (ICSGRC), Shah Alam, Malaysia, 23 July 2022; IEEE: New York, NY, USA, July 23, 2022; pp. 120–125.
9. Matic, P.; Bego, O.; Goić, R. A Systematic Approach to a Time Series Neural Model Development for River Flow Forecasting. *Int. Rev. Autom. Control* **2012**, *5*, 367–372.
10. Demuth, H.; Beale, M.; Hagan, M. *Neural Network Toolbox^{TM} 6 User's Guide*; The MathWorks Inc.: Natick, MA, USA, 1992.
11. Salehi, A.; Montazeri-Gh, M. Black Box Modeling of a Turboshaft Gas Turbine Engine Fuel Control Unit Based on Neural NARX. *Proc. Inst. Mech. Eng. Part M J. Eng. Marit. Environ.* **2019**, *233*, 949–956. [CrossRef]
12. Boussaada, Z.; Curea, O.; Remaci, A.; Camblong, H.; Mrabet Bellaaj, N. A Nonlinear Autoregressive Exogenous (NARX) Neural Network Model for the Prediction of the Daily Direct Solar Radiation. *Energies* **2018**, *11*, 620. [CrossRef]
13. Cybenko, G. Approximation by Superpositions of a Sigmoidal Function. *Math. Control Signals Syst.* **1989**, *2*, 303–314. [CrossRef]
14. De Vos, N.J.; Rientjes, T.H.M. Hydrology and Earth System Sciences Constraints of Artificial Neural Networks for Rainfall-Runoff Modelling: Trade-Offs in Hydrological State Representation and Model Evaluation. *Hydrol. Earth Syst. Sci.* **2005**, *9*, 111–126. [CrossRef]
15. Yadav, S.; Shukla, S. Analysis of K-Fold Cross-Validation over Hold-Out Validation on Colossal Datasets for Quality Classification. In Proceedings of the Proceedings—6th International Advanced Computing Conference, IACC 2016, Bhimavaram, India, 27–28 February 2016; Institute of Electrical and Electronics Engineers Inc.: New York, NY, USA, August 16, 2016; pp. 78–83.
16. Kuhn, M.; Johnson, K. *Applied Predictive Modeling*; Springer: New York, NY, USA, 2013; ISBN 978-1-4614-6848-6.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.