

# Robustness and Generalization Performance of Deep Learning Models on Cyber-Physical Systems: A Comparative Study

Alexander Windmann, Henrik Steude and Oliver Niggemann

Helmut Schmidt University, Hamburg, Germany

{alexander.windmann, henrik.steude, oliver.niggemann}@hsu-hh.de

## Abstract

Deep learning (DL) models have seen increased attention for time series forecasting, yet the application on cyber-physical systems (CPS) is hindered by the lacking robustness of these methods. Thus, this study evaluates the robustness and generalization performance of DL architectures on multivariate time series data from CPS. Our investigation focuses on the models' ability to handle a range of perturbations, such as sensor faults and noise, and assesses their impact on overall performance. Furthermore, we test the generalization and transfer learning capabilities of these models by exposing them to out-of-distribution (OOD) samples. These include deviations from standard system operations, while the core dynamics of the underlying physical system are preserved. Additionally, we test how well the models respond to several data augmentation techniques, including added noise and time warping. Our experimental framework utilizes a simulated three-tank system, proposed as a novel benchmark for evaluating the robustness and generalization performance of DL algorithms in CPS data contexts. The findings reveal that certain DL model architectures and training techniques exhibit superior effectiveness in handling OOD samples and various perturbations. These insights have significant implications for the development of DL models that deliver reliable and robust performance in real-world CPS applications.

## 1 Introduction

Cyber-physical systems (CPS) are complex systems that integrate physical and computational components, such as autonomous vehicles, power grids, and industrial control systems [Niggemann *et al.*, 2023]. These systems must operate reliably and accurately in dynamic and unpredictable environments, where the consequences of failure can be severe. Deep learning (DL) has shown great potential for improving the performance and robustness of CPS by enabling systems to learn from data and adapt to changing conditions [Khan *et al.*, 2021]. However, the deployment of DL algorithms

in CPS is not without challenges, as they must operate efficiently in real-time, handle large amounts of data, and provide accurate predictions despite noisy or adversarial input and varying environmental conditions.

One of the critical challenges in deploying DL algorithms in CPS is the ability to generalize to new and unseen scenarios [Bansal *et al.*, 2021]. In real-world applications, the distribution of the data may change slowly over time or suddenly if unexpected events occur, yet the models must be able to adapt to these changes and maintain their accuracy and reliability. Furthermore, in contrast to other application areas of DL like computer vision or natural language processing, DL models in CPS generally have to learn from a domain-specific dataset which only contains sensor data from one system [Zimmering *et al.*, 2021]. Once this system changes, e.g. because a sensor is added, usually the whole DL model has to be retrained from scratch. What is needed is a model that generalizes from the initial dataset, i.e. it can transfer learned knowledge about the physical systems to an altered system or a different application.

In this paper, we focus on evaluating the systematic generalization performance of several DL algorithms on a simulation of a three-tank system which follows a sequence of process phases, changing the in- and outflow of the tanks in a predictable manner. This system presents several challenges for DL algorithms, including nonlinear dynamics, ad-hoc process phase changes, long-term trends and noisy sensor measurements. We test the algorithms on new scenarios that they have not seen during training to assess their ability to capture system dynamics, learn from limited data and handle out-of-distribution (OOD) samples.

The research questions (RQ) to be discussed in this paper can be summarized as follows:

1. What deep learning model architectures demonstrate robustness to perturbations in CPS, specifically noise, variation on timings and faulty sensors, and how much do these scenarios impact the models' performance?
2. How effectively can these DL model architectures generalize to unfamiliar system behaviors, which may include systems operating at different scales or presenting new combinations of process phases, all while ensuring the fundamental system dynamics are maintained?
3. Which DL model architectures exhibit the highest capa-

bility to transfer system-specific learned knowledge to related tasks following fine-tuning?

4. To what extent do data augmentation techniques during training improve the robustness and generalization capabilities of different DL model architectures?

We hope that this study can help to uncover the key design principles and modeling approaches that can improve the robustness and reliability of DL models in CPS, especially when dealing with OOD samples. Furthermore, by making our code and the simulation environment available openly, we provide a benchmark for researchers of robust DL models on CPS.<sup>1</sup>

The structure of this paper is as follows: Section 2 surveys relevant work, underscoring the research gaps this study aims to bridge. Section 3 describes our simulation environment, the selected models, and our comprehensive experimental setup. Section 4 presents the outcomes of our experiments alongside a detailed analysis. In Section 5, we summarize our findings, draw conclusions, and propose potential avenues for future research.

## 2 Related Work

Due to the rising complexity of CPS, there is an increasing demand on using DL in industrial applications, such as anomaly detection [Luo *et al.*, 2021], predictive maintenance [de Beaulieu *et al.*, 2022] and planning [Köcher *et al.*, 2022]. As CPS share many similarities with dynamical systems, most DL models that are in use focus on processing time series data. Such models include recurrent networks like LSTM networks [Hochreiter and Schmidhuber, 1997; Chen *et al.*, 2022] and convolutional models, such as Temporal Convolutional Networks (TCNs) [Lea *et al.*, 2017; Meng *et al.*, 2020]. Recently, Transformers [Vaswani *et al.*, 2017; Geneva and Zabarar, 2022] and graph neural networks [Kipf *et al.*, 2018; Wu *et al.*, 2020] have also been gaining traction in modeling physical systems.

Robustness is a fundamental aspect of DL methods, as it directly impacts the reliability and safety of applications [Bengio *et al.*, 2021]. Despite their effectiveness in narrow tasks, DL models often exhibit limited performance when faced with small perturbations in the input data, such as adversarial attacks [Tsipras *et al.*, 2019; Buzhinsky *et al.*, 2021]. This limitation presents a significant challenge in CPS, where robustness is paramount for the safe deployment of DL methods [Khan *et al.*, 2021]. Particularly, CPS applications, given their diverse sensory input, are also prone to adversarial attacks [Li *et al.*, 2020]. Despite this, a comprehensive study on the impact of various perturbations in CPS on DL architectures is yet to be conducted.

OOD generalization is another crucial aspect in DL. Most models rely on the assumption of independent and identically distributed data and struggle to generalize to new, previously unseen data [Schölkopf *et al.*, 2021]. When the distribution of input data deviates significantly from the training data, the performance of the models can deteriorate.

<sup>1</sup>Code: <https://github.com/alexwndm/robust-ml-for-cps>

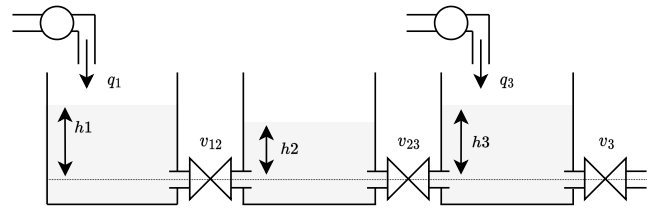


Figure 1: This figure schematically shows the three-tank system used in the experiments. It includes the unobservable inflow  $q_1$  and  $q_3$ , the valve coefficients  $v_{12}$ ,  $v_{23}$  and  $v_3$  as well as the observable water levels  $h_1$ ,  $h_2$  and  $h_3$ .

For dynamical systems, learning the underlying causal variables is one approach [Lippe *et al.*, 2022a; Lippe *et al.*, 2022b], which has proven successful in CPS data applications such as prognostics and health management [Koutroulis *et al.*, 2022]. Another research direction explores the learning of independent mechanisms from data [Goyal *et al.*, 2021; Goyal *et al.*, 2022], often resulting in modular model architectures that exhibit improved OOD predictions on CPS data [Bansal *et al.*, 2021]. However, a comprehensive empirical study of the generalization performance of DL models for CPS remains an open area for exploration.

Data augmentation is one of several learning techniques that has shown to improve robustness of DL models [Shorten and Khoshgoftaar, 2019]. Beyond computer vision, it has found success in areas such as time series analysis [Wen *et al.*, 2021], Natural Language Processing (NLP) [Bayer *et al.*, 2022], and graph machine learning [Zhao *et al.*, 2023]. Recent years have seen the application of data augmentation in CPS as well [Yin *et al.*, 2021]. There is, however, still a lack of works that systematically investigate the impact of data augmentation on CPS applications.

## 3 Methodology

In this study, we simulate a three-tank system, as depicted in Figure 1, to evaluate the robustness and generalization performance of various DL architectures on CPS. Our primary focus is on forecasting, a task that necessitates a deep understanding of system dynamics. Forecasting is not only analogous to various CPS-related tasks, such as industrial process control [Downs and Vogel, 1993] and prognostics and health management [Lee *et al.*, 2014], but can also serve as a self-supervised learning pre-training task that could benefit other applications [Zerveas *et al.*, 2021].

The transparent system dynamics offered by the three-tank simulation provide full control over system behavior, enabling qualitative evaluation through minor adjustments. Furthermore, a simulation allows to evaluate the models' generalization performance systematically. In general, evaluating the generalization performance of forecasting models is challenging, since predicting arbitrary time series is infeasible. However, simulations of CPS like the three-tank system share a common foundation in the underlying physical and dynamical system. This enables the generation of realistic OOD samples that serve as reasonable targets for testing generalization performance. Moreover, our simulation can be readily

adapted to more complex scenarios, such as a four-tank system, thereby extending its applicability in studying DL model performance across a diverse range of CPS tasks. Openly accessible data from real-life CPS lacks this flexibility, as the dataset is typically fixed and intricate to the extent that the impacts of interventions cannot be reliably predicted during experiment design. Therefore, our simulation-based approach provides a more controlled and adaptable framework for understanding and enhancing the performance of DL models in CPS applications.

### 3.1 Simulation Environment

To generate the training dataset, we employ a systematic approach, filling and mixing the tanks in a controlled and predictable manner. The dataset is composed of subsequential process phases, each with a duration of 50 time steps. During each process phase, control variables, such as inflow rates and valve coefficients, are held constant, allowing the system’s dynamics to evolve according to the predefined underlying equations:

$$\begin{aligned}\dot{h}_1 &= q_1 - k_{v12} \text{sign}(h_1 - h_2) \sqrt{|h_1 - h_2|} \\ \dot{h}_2 &= k_{v12} \text{sign}(h_1 - h_2) \sqrt{|h_1 - h_2|} \\ &\quad - k_{v23} \text{sign}(h_2 - h_3) \sqrt{|h_2 - h_3|} \\ \dot{h}_3 &= q_3 + k_{v23} \text{sign}(h_2 - h_3) \sqrt{|h_2 - h_3|} - k_{v3} \sqrt{h_3}\end{aligned}\quad (1)$$

where  $h_1(t)$ ,  $h_2(t)$  and  $h_3(t)$  describe the water levels of the three tanks at the time step  $t$ ,  $q_1$  and  $q_3$  describe the inflow and  $k_{v12}$ ,  $k_{v23}$  and  $k_{v3}$  describe the valve coefficients as depicted in Figure 1.

To create a closer resemblance to real-world CPS, we simulate production by progressing through a defined sequence of process phases: initially filling Tank 1, followed by mixing Tanks 1 and 2, then mixing Tanks 2 and 3, refilling Tank 1, repeating the two mixing stages, and ultimately emptying Tank 3. To emulate a more realistic inflow pattern, we introduce a long-term cosine trend, representing the ramp-up and shutdown of inflow rates. Additionally, Gaussian noise is incorporated into sensor values to further increase the simulation’s realism. Detailed information regarding the simulation and dataset generation can be found in the associated repository, and a representative sample is illustrated in Figure 2.

### 3.2 Models Under Test

In this study, we evaluate the robustness and generalization performance of various deep learning models using a three-tank system as the test environment. Guided by foundational architectural principles widely adopted in the DL research community – recurrence, convolutions, and attention mechanisms – we select a prominent model from each category. Furthermore, we experiment with different model variations within each category. The models under investigation include:

1. Multilayer Perceptron (MLP): As a baseline, we employ an MLP with up to three hidden layers, utilizing the Rectified Linear Unit (ReLU) activation function between layers.

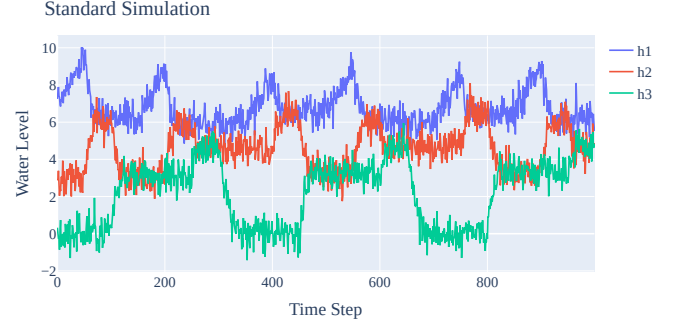


Figure 2: This figure shows a section from the ‘standard’ scenario dataset of the three-tank simulation. A sample consists of 250 time steps, the forecasting target is 50 time steps long.

2. Gated Recurrent Unit (GRU) [Cho *et al.*, 2014]: To evaluate recurrent architectures, we implement a GRU network. We adopt two forecasting approaches: (i) using a linear layer on the final hidden state to predict the entire time series directly, and (ii) employing an auto-regressive method, denoted as GRU-AR, to predict the target time series step by step.
3. Temporal Convolutional Network (TCN) [Lea *et al.*, 2017]: To examine convolutional networks, we implement a TCN. We experiment with significantly reducing the dimensionality in the latent space using the autoencoder framework [Bengio *et al.*, 2013] similar to TCN-AEs [Meng *et al.*, 2020] additional to a decoder for the forecasting objective. We refer to this as the TCN-Forecasting-Autoencoder (TCN-FAE).
4. Transformer [Vaswani *et al.*, 2017]: To assess attention networks, we implement a vanilla encoder-decoder Transformer. Additionally, we explore embedding the time series using causal convolutions, as proposed by [Li *et al.*, 2019]. This is denoted as Transformer-CE.

### 3.3 Experimental Setup

The backbone of all subsequent experiments are models trained on a dataset derived from the three-tank simulation delineated in Subsection 3.1. This time series is partitioned into training, validation, and testing intervals. For the training dataset, we randomly select 1000 samples from the training interval, while the validation and testing datasets each comprise 100 samples. Each model is fed a random 250-time-step interval as input and assigned the task of forecasting the subsequent 50 time steps of the three water levels. Upon the completion of each epoch, we compute the Mean Squared Error (MSE) on the validation dataset. The model manifesting the lowest epoch validation loss across all models and epochs is chosen during hyperparameter tuning. All models employ the Adam optimizer [Kingma and Ba, 2015], with the learning rate halved if the epoch validation loss does not reduce for 25 epochs [Bottou, 2012]. Although all models are trained for a maximum of 1000 epochs, early stopping is enforced if the validation loss ceases to decrease for 50 subsequent epochs. Comprehensive implementation details, including hyperparameters and training procedures, are available in the asso-

ciated repository.

It should be noted that the samples from the training, validation, and testing datasets, drawn from the same simulation, bear a similar distribution. This contrasts with the forthcoming experiments, where models predict samples drawn from alternative simulations, termed as scenarios. The time series referenced in Subsection 3.1 will henceforth be referred to as the 'standard' scenario.

### Experiment 1: Robustness to Perturbations

To address RQ 1, which pertains to the robustness of the models in the face of perturbations, we evaluate the trained models on various simulations where the time series are influenced by specific perturbations. The scenarios are as follows:

- **Scenario 1: Sensor Faults.** We introduce anomalies to the models' input data to simulate sensor faults, a common issue in real-life CPS. In half of the instances, we add a point anomaly for one sensor at a random time step. For the remaining instances, we simulate a dead sensor by setting its value to zero for up to 50 time steps. We ensure that these anomalies are excluded from the last input steps and the target sequence, thereby evaluating each model on an accurate time series.
- **Scenario 2: Increased Noise.** We simulate the three-tank system with thrice the amount of Gaussian noise on all sensor values, making the forecasting task more challenging.
- **Scenario 3: Variable Process Phase Duration.** We manipulate the duration of different process phases, such as filling or mixing tanks, by adding noise to the 50 step process phase duration interval in the simulation.

### Experiment 2: OOD Generalization

To assess the generalization performance of various DL models on OOD samples and thereby answer RQ 2, we investigate their forecasting performance in different scenarios where the behavior or the settings of the underlying three-tank system change. The underlying causal and physical system described in Equation (1) serves as the foundation for each simulation, ensuring that OOD samples are challenging yet grounded in reality. A DL model that fully captures the underlying causal and dynamic system should exhibit improved forecasting performance in these new scenarios.

The scenarios are as follows:

- **Scenario 4: Scaled Inflow and Outflow.** We increase both the inflow and valve coefficients by 50%. This results in generally higher water levels while preserving the system's behavior.
- **Scenario 5: Independent Process Phase Merging.** We simultaneously fill Tank 1 and empty Tank 3, combining two independent process phases.
- **Scenario 6: Independent Process Phase Merging with Additional Stable Process Phase.** By combining the two process phases in the previous scenario, we effectively reduce the underlying cycle from 7 to 6 process phases. To assess the effect of this change, we add an additional process phase to the cycle during which the water levels remain unchanged.

- **Scenario 7: Dependent Process Phase Merging.** We create a novel process phase by opening both valves between the tanks simultaneously, mixing all tanks at once.

Additionally, to answer RQ 3 we try to evaluate the transfer learning capabilities of the models by fine-tuning them on the new scenarios. We do this by reusing the model weights and retraining the models on a new objective: predict the next 50 time steps of the water tanks of the new OOD scenario. We limit the maximum number of epochs the models are allowed to train to 50 and choose the model checkpoint with the best loss on the validation dataset of the OOD scenario. Models that have better learned the underlying concepts, such as mixing or filling of tanks, should adapt more quickly when fine-tuned to the new system.

### Experiment 3: Data Augmentation

To investigate the extent to which the models described in Subsection 3.2 can benefit from data augmentation techniques (addressing RQ 4), we select the models that have been trained under the 'standard' scenario and train them on additional augmented data. This augmented data was generated by applying two types of transformations on the original dataset derived from the 'standard' scenario:

1. **Added Noise:** We enhance the time series by adding Gaussian noise. This additive distortion complements the inherent noise that forms part of the simulation.
2. **Time Warping:** We introduce variability in the time series by adding random noise to the original time step intervals, thereby warping the time scale. The new values are obtained by interpolating the original time series values at these adjusted time steps and are subsequently mapped to the original time indices. The resultant time series maintains similarity to the original, albeit with more randomized timings.

The models pre-trained under the 'standard' scenario are then fine-tuned on these synthetically augmented datasets following an approach akin to that adopted in Experiment 2.

## 4 Results

### 4.1 Experiment 1: Robustness to Perturbations

The results of the first experiment are outlined in Table 1. In the 'standard' scenario, all model architectures demonstrated the capability to accurately predict the future water level of the three tanks, with minimal variation in predictive accuracy. Notably, the GRU model with an auto-regressive forecasting pattern generated the most precise predictions. Interestingly, more sophisticated architectures like TCNs or Transformers did not surpass the predictive performance of the MLP.

The good performance of the MLP is further evident in the scenarios involving perturbations introduced to the three-tank simulation. Particularly in the presence of sensor faults, the MLP outperformed all other models, despite its performance drop. For the simulation with faulty sensors, a clear pattern emerged across other architectures: recurrent architectures are affected by these perturbations, convolutional models performed worse and Transformer variants utterly failed to predict the time series when faced with faulty data, which partly answers RQ 1.

Model	Standard	Scn. 1	Scn. 2	Scn. 3
MLP	0.32	<b>0.98</b>	<b>2.34</b>	<b>0.72</b>
GRU	0.33	2.97	2.48	0.92
GRU-AR	<b>0.28</b>	2.05	2.42	0.89
TCN	0.38	5.65	2.77	1.21
TCN-FAE	0.34	4.02	2.40	1.11
Transformer	<u>0.38</u>	<u>15.55</u>	<u>3.78</u>	<u>1.28</u>
Transf.-CE	0.37	8.78	3.40	1.04

Table 1: Performance comparison of models across different scenarios (MSE). Lower values indicate better performance. The best performance for each scenario is highlighted in bold, the worst is underlined. The scenarios include: ‘sensor faults’ (Scn. 1), ‘increased noise’ (Scn. 2) and ‘variable process phase duration’ (Scn. 3).

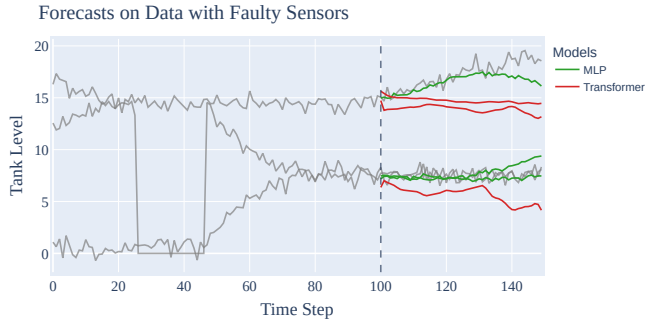


Figure 3: Sample forecast for selected models on the simulation with faulty sensors. While the MLP is mostly unaffected, the Transformer completely fails to predict the future time steps.

Figure 3 provides a graphical representation of the MLP’s and Transformer’s forecast when presented with a faulty sensor. While the MLP’s forecast was minimally affected, the Transformer faltered significantly in its predictions. It incorrectly predicted the water levels of all three tanks. Notably, it predicted the water level of Tank 2 to start at a high level, even though the tank has just been emptied. This underlines the model’s deficiency in learning fundamental properties of the underlying physical system, as such sudden fluctuations of water levels are implausible in the standard three-tank scenario the model was trained on. As this performance drop has not been foreseeable during the ‘standard’ scenario, the experiment highlights the importance of performing extensive tests on models before they are applied in production on a real-life CPS.

In the scenarios involving heightened noise and variable process phase durations, the behavior of the models is similar to Scenario 1, albeit with less pronounced differences. Given the increased uncertainty in Scenarios 2 and 3, a rise in MSE was anticipated as a perfect prediction of such randomness is inherently infeasible. Despite this, the MLP maintained its superior performance. Transformers, on the other hand, demonstrated inferior performance under increased noise, while recurrent models outperformed other architectures, barring the MLP, under variable process phase durations, which answers RQ 1.

In summary, an increased complexity in the model archi-

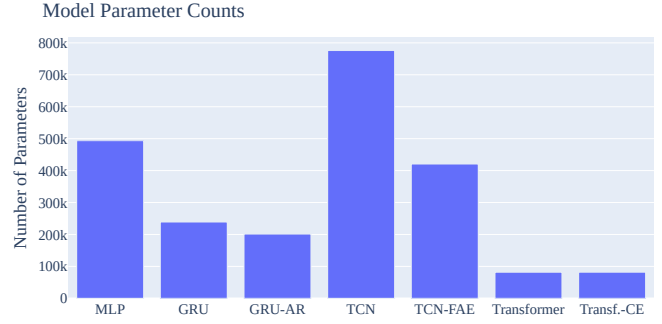


Figure 4: This figure shows the parameter count for the tested models.

ture can lead to impaired robustness to data perturbations, as exemplified by the MLP’s performance in the tested simulations. However, no distinct correlation between model size and robustness was discernible across model architectures, as depicted in Figure 4. Hyperparameters for all models were determined through extensive grid search, as detailed in Section 3.3. It was observed that larger convolutional architectures such as TCNs tended to perform better, while recurrent models were relatively smaller. Transformers, generally, have been noted to perform better when scaled up [Brown *et al.*, 2020]. However, due to hardware constraints, we were limited by the self-attention algorithm’s memory and runtime of  $O(n^2)$  for an input sequence length of  $n$ . Given the input length of 250, even relatively small Transformer variants pushed the A100 GPU used for model testing to its limits. This restriction, however, was universally applicable to all models, ensuring a level playing field. It would be intriguing to examine the performance of more recent Transformer variants with improved runtime and memory footprint when subjected to similar scenarios.

## 4.2 Experiment 2: OOD Generalization

In the subsequent experiment, we assessed the models’ performance across a variety of scenarios where the behavior of the underlying three-tank system was altered. The findings of this investigation are detailed in Table 2.

In Scenario 4, both the inflow and the valve coefficients of the system were increased by 50%. Despite the similar behavior to the learned ‘standard’ scenario, the sensor values were generally elevated. There is a clear distinction in between the MLP and the TCN variants, which yielded satisfactory predictions, and the GRU and Transformer variants, which did not meet expectations. Following a fine-tuning phase of 50 epochs, a significant improvement was noted in most models. The exceptions are the TCN-FAE, which shows a decreasing performance, and the MLP, which already demonstrated excellent performance prior to fine-tuning.

Scenario 5 encompassed two process phases that did not directly influence each other, executed concurrently. This implied that models with the ability to integrate the learned behavior of two separate process phases would outperform others. Regrettably, none of the models predicted the new scenario accurately without the benefit of fine-tuning. Curiously, the auto-regressive GRU variant outperforms the other

Model	Scenario 4		Scenario 5		Scenario 6		Scenario 7	
	Epoch 0	Epoch 50	Epoch 0	Epoch 50	Epoch 0	Epoch 50	Epoch 0	Epoch 50
MLP	<b>0.53</b>	<b>0.53</b>	6.01	0.44	<b>1.27</b>	0.43	<b>6.02</b>	<b>0.35</b>
GRU	10.27	<b>0.53</b>	4.79	<b>0.35</b>	1.28	0.39	<u>9.65</u>	0.37
GRU-AR	14.80	<b>0.53</b>	<b>2.56</b>	<b>0.35</b>	1.32	<b>0.35</b>	8.12	0.36
TCN	1.52	1.32	5.18	0.57	<u>2.52</u>	0.50	8.28	1.02
TCN-FAE	1.28	1.49	<u>6.47</u>	<u>2.93</u>	1.71	<u>1.40</u>	7.21	<u>2.19</u>
Transformer	13.06	<u>2.59</u>	4.13	0.47	1.51	0.46	9.45	0.99
Transf.-CE	<u>19.04</u>	1.66	4.97	0.44	1.57	0.40	8.06	0.79

Table 2: Comparative performance of different models fine-tuned across various test scenarios. The table presents the MSE values obtained from models trained on the ‘standard’ scenario and subsequently fine-tuned for 0 or 50 epochs on the respective scenario. Lower MSE values signify better performance. For each scenario and fine-tuning duration (0 or 50 epochs), the highest performing model is denoted in bold, while the lowest performing model is underlined. The scenarios include: ‘scaled inflow and outflow’ (Scenario 4), ‘independent process phase merging’ (Scenario 5), ‘independent process phase merging with an additional stable phase’ (Scenario 6), and ‘dependent process phase merging’ (Scenario 7).

models before and after fine-tuning. It would be interesting to further investigate the effect of this forecasting method on integrating learned behavior.

The merging of two scenarios introduced a new complexity: the system had to transition from cycling through a fixed number of process phases to one less phase. To explore the extent to which the models rely on this cyclic structure in their predictions, an additional ‘stable’ process phase was incorporated into Scenario 6. Hypothetically, the ‘stable’ process phase where the water levels in the tanks remain constant presents a new challenge for the models, as it is not part of the ‘standard’ scenario the models have been trained on. However, Table 2 reveals that the initial MSE values, prior to fine-tuning, were substantially lower than in Scenario 5. This suggests that the models have, to a significant degree, learned the underlying cyclic structure of the simulation. After fine-tuning, this disparity disappeared, indicating that the models are capable of swiftly adapting to changes in the cycle length.

Finally, Scenario 7 involved the merger of two dependent process phases where Tanks 1 and 2 were mixed while Tank 2 was concurrently mixed with Tank 3. An effective prediction of this scenario without fine-tuning would necessitate a model capable of understanding the ‘mixing tanks’ mechanism and applying it simultaneously from two perspectives. As anticipated, the models struggled more in this scenario than in Scenario 5. The MLP was the only model that managed to maintain a somewhat consistent performance, albeit with a high MSE.

In a comprehensive review of all scenarios, it is evident that the majority of models fail to predict the correct OOD behavior initially, but demonstrate a robust ability to adapt to the new system dynamics with fine-tuning, which answers RQ 2 and 3. However, the TCN-FAE model stands as an exception, often failing to show improvement within the given epoch limit. It is also the sole model that exhibits a performance decline in one scenario after fine-tuning for 50 epochs. A common trend observed across the tested models is an initial degradation in performance upon fine-tuning all model weights, as illustrated in Figure 5. Nevertheless, for most models, this loss quickly reverses, while the TCN-FAE often

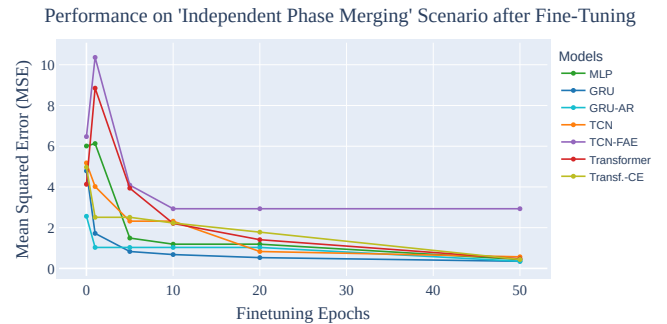


Figure 5: Performance of different models on the scenario with independent phase merging after fine-tuning.

stagnates, hitting a performance plateau. Despite its ability to produce accurate forecasts on the ‘standard’ scenario, the TCN-FAE does not seem optimally designed for a transfer learning scenario, as highlighted by this experiment. While further fine-tuning could potentially enhance the model’s performance, it does not appear to leverage the system knowledge to the same degree as the other models. This could potentially be attributed to the bottleneck in the latent dimension or the autoencoder objective, which is learned concurrently with the predictions. Further investigations into these potential factors might provide more insight into the limitations observed in the TCN-FAE model’s performance across various scenarios.

### 4.3 Experiment 3: Data Augmentation

In this experiment, we aim to assess how effectively the models adapt to data augmentation techniques. The results are presented in Table 3. As anticipated, both data augmentation techniques marginally reduce the performance on the ‘standard’ scenario on which the models were initially trained. Given that the principal motivation for employing data augmentation is to enhance model robustness, we examined performance in a scenario involving a perturbation (Scenario 1) and another reflecting OOD system behavior (Scenario 5).

The introduction of noise to the time series appears ben-



Model	Standard			Scenario 1			Scenario 5		
	None	Noise	Warp	None	Noise	Warp	None	Noise	Warp
MLP	0.32	0.34	0.47	<b>0.98</b>	<b>0.95</b>	<b>1.33</b>	6.01	6.31	3.98
GRU	0.33	0.40	0.49	2.97	2.98	3.59	4.79	5.64	<b>2.03</b>
GRU-AR	<b>0.28</b>	<b>0.30</b>	<b>0.46</b>	2.05	1.53	1.88	<b>2.56</b>	<b>3.10</b>	2.48
TCN	<u>0.38</u>	0.39	0.64	5.65	2.97	3.48	5.18	5.08	3.39
TCN-FAE	0.34	0.43	0.51	4.02	4.37	2.83	<u>6.47</u>	<u>6.43</u>	2.10
Transformer	<u>0.38</u>	0.54	<u>0.74</u>	<u>15.55</u>	<u>9.40</u>	11.58	4.13	5.48	2.92
Transf.-CE	0.37	<u>0.67</u>	0.64	8.78	7.66	<u>12.06</u>	4.97	5.26	<u>3.99</u>

Table 3: Performance comparison of different models fine-tuned on data augmented with various techniques. This table showcases the MSE values derived from models initially trained on the ‘standard’ scenario and subsequently fine-tuned using the respective data augmentation techniques (added noise or time warping). Lower MSE values indicate better performance. For each scenario and data augmentation technique, the best performing model is highlighted in bold, while the least performing model is underlined. Scenarios encompass the ‘standard’ scenario, ‘sensor faults’ (Scenario 1), and ‘independent process phase merging’ (Scenario 5).

eficial when managing perturbations such as faulty sensors, as most models exhibit improved performance on Scenario 1. However, this trend is not universally observed, as a few models display slightly diminished performance. Given that the performance on scenarios with OOD behavior and the ‘standard’ scenarios is compromised for most models, the indiscriminate application of this data augmentation technique for all model architectures is not recommended.

The impact of time warping diverges from the other data augmentation technique. While it generally enhances performance in the face of perturbations, the inclusion of noise proves more advantageous in Scenario 1. Nevertheless, time warping substantially improves performance when confronted with OOD behavior in Scenario 5 for all models. However, this performance enhancement comes with a trade-off: a more pronounced performance decline on the ‘standard’ scenario compared to the other data augmentation technique.

To answer RQ 4, we analyzed the individual models responses to data augmentation techniques. The MLP, demonstrating good performance across most scenarios, doesn’t derive as much benefit from data augmentation as other models, suggesting cautious application. The same applies to the recurrent models, with the exception of the GRU with an autoregressive forecasting method, which significantly enhances its predictions when dealing with sensor faults. Convolutional methods and models utilizing attention mechanisms, initially underperforming, mostly see improved robustness from data augmentation techniques, particularly the TCN variants. The TCN-FAE model responds exceptionally well to time warping. This trend warrants further investigation, necessitating a more extensive study.

## 5 Conclusion and Future Work

This paper aimed to assess the robustness and generalization performance of various DL architectures applied to CPS data. The evaluated simulation indicated that recurrent architectures demonstrated robustness against perturbations, while convolutional methods exhibited the ability to generalize to certain OOD scenarios. Notably, MLP exhibited impressive performance in both areas. Conversely, Transformer models

showed significant susceptibility to perturbations and struggled with generalization. However, this observation must be tempered by the fact that hardware limitations restricted the size and complexity of the Transformer models we could train in this study. Most models demonstrated a successful transfer of knowledge to related tasks after fine-tuning, particularly the recurrent architectures and the MLP, with a noted exception being a specific TCN variant.

The findings of this study carry important implications for the application of DL methods. Firstly, practitioners must approach the application of DL models in CPS with caution, as performance disparities only became apparent when subjecting the models to specific test scenarios. Secondly, our study indicates that even relatively simple models, like MLPs, can perform surprisingly well with time series data. Lastly, despite most models initially struggling with generalization, fine-tuning proved to be effective. This suggests that fine-tuning models from related CPS tasks might be more cost-effective than training new models from scratch.

Our study also has several limitations. The underlying three-tank system is relatively simple and predictable. Future work could enhance the realism of the CPS simulation by incorporating control loops and discrete sensor values. As an alternative, the experiments detailed within this paper could be conducted using a physical testbed, adding another layer of practical validation. Additionally, while the models tested provide a broad overview of DL architectures, newer variants that address existing limitations should be assessed to present a more accurate representation of the current state-of-the-art. Finally, alleviating hardware constraints would enable testing of more complex models, like large Transformers, thereby harnessing their full potential. Future research should aim to address these limitations and investigate more robust and generalizable DL architectures, such as causal or modular models.

Through this study, we have gained valuable insights into the application of DL models on CPS data. Our findings pave the way for more robust and reliable real-world CPS applications, thus contributing to the continual advancement of this domain.

## Acknowledgments

This research paper is funded by dtcc.bw – Digitalization and Technology Research Center of the Bundeswehr.

## References

- [Bansal *et al.*, 2021] Hritik Bansal, Gantavya Bhatt, Pankaj Malhotra, and Prathosh AP. Systematic Generalization in Neural Networks-based Multivariate Time Series Forecasting Models. In *2021 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, July 2021.
- [Bayer *et al.*, 2022] Markus Bayer, Marc-André Kaufhold, and Christian Reuter. A Survey on Data Augmentation for Text Classification. *ACM Computing Surveys*, 55(7):146:1–146:39, December 2022.
- [Bengio *et al.*, 2013] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: a review and new perspectives. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(8):1798–1828, August 2013.
- [Bengio *et al.*, 2021] Yoshua Bengio, Yann Lecun, and Geoffrey Hinton. Deep learning for AI. *Communications of the ACM*, 64(7):58–65, June 2021.
- [Bottou, 2012] Léon Bottou. Stochastic gradient descent tricks. In Grégoire Montavon, Geneviève B Orr, and Klaus-Robert Müller, editors, *Neural Networks: Tricks of the Trade: Second Edition*, pages 421–436. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.
- [Brown *et al.*, 2020] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language Models are Few-Shot Learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc., 2020.
- [Buzhinsky *et al.*, 2021] Igor Buzhinsky, Arseny Nerinovsky, and Stavros Tripakis. Metrics and methods for robustness evaluation of neural networks with generative models. *Machine Learning*, July 2021.
- [Chen *et al.*, 2022] Chuang Chen, Ningyun Lu, Bin Jiang, and Yin Xing. A Data-Driven Approach for Assessing Aero-Engine Health Status. *IFAC-PapersOnLine*, 55(6):737–742, January 2022.
- [Cho *et al.*, 2014] Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar, October 2014. Association for Computational Linguistics.
- [de Beaulieu *et al.*, 2022] Martin Hervé de Beaulieu, Mayank Shekhar Jha, Hugues Garnier, and Farid Cerbah. Unsupervised Remaining Useful Life Prediction through Long Range Health Index Estimation based on Encoders-Decoders. *IFAC-PapersOnLine*, 55(6):718–723, January 2022.
- [Downs and Vogel, 1993] J. J. Downs and E. F. Vogel. A plant-wide industrial process control problem. *Computers & Chemical Engineering*, 17(3):245–255, March 1993.
- [Geneva and Zabarar, 2022] Nicholas Geneva and Nicholas Zabarar. Transformers for modeling physical systems. *Neural Networks*, 146:272–289, February 2022.
- [Goyal *et al.*, 2021] Anirudh Goyal, Alex Lamb, Jordan Hoffmann, Shagun Sodhani, Sergey Levine, Yoshua Bengio, and Bernhard Schölkopf. Recurrent Independent Mechanisms. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021.
- [Goyal *et al.*, 2022] Anirudh Goyal, Aniket Rajiv Didolkar, Alex Lamb, Kartikeya Badola, Nan Rosemary Ke, Nasim Rahaman, Jonathan Binas, Charles Blundell, Michael Curtis Mozer, and Yoshua Bengio. Coordination Among Neural Modules Through a Shared Global Workspace. In *International Conference on Learning Representations*, January 2022.
- [Hochreiter and Schmidhuber, 1997] Sepp Hochreiter and Jürgen Schmidhuber. Long Short-Term Memory — Neural Computation. <https://dl.acm.org/doi/10.1162/neco.1997.9.8.1735>, 1997.
- [Khan *et al.*, 2021] Samir Khan, Seiji Tsutsumi, Takehisa Yairi, and Shinichi Nakasuka. Robustness of AI-based prognostic and systems health management. *Annual Reviews in Control*, 51:130–152, January 2021.
- [Kingma and Ba, 2015] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.
- [Kipf *et al.*, 2018] Thomas Kipf, Ethan Fetaya, Kuan-Chieh Wang, Max Welling, and Richard Zemel. Neural Relational Inference for Interacting Systems. In *Proceedings of the 35th International Conference on Machine Learning*, pages 2688–2697. PMLR, July 2018.
- [Köcher *et al.*, 2022] Aljosha Köcher, René Heesch, Niklas Widulle, Anna Nordhausen, Julian Putzke, Alexander Windmann, and Oliver Niggemann. A Research Agenda for AI Planning in the Field of Flexible Production Systems. In *2022 IEEE 5th International Conference on Industrial Cyber-Physical Systems (ICPS)*, pages 1–8, May 2022.
- [Koutroulis *et al.*, 2022] Georgios Koutroulis, Belgin Mutlu, and Roman Kern. Constructing robust health indicators



- from complex engineered systems via anticausal learning. *Engineering Applications of Artificial Intelligence*, 113:104926, August 2022.
- [Lea *et al.*, 2017] Colin Lea, Michael D. Flynn, Rene Vidal, Austin Reiter, and Gregory D. Hager. Temporal Convolutional Networks for Action Segmentation and Detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 156–165, 2017.
- [Lee *et al.*, 2014] Jay Lee, Fangji Wu, Wenyu Zhao, Masoud Ghaffari, Linxia Liao, and David Siegel. Prognostics and health management design for rotary machinery systems—Reviews, methodology and applications. *Mechanical Systems and Signal Processing*, 42(1):314–334, January 2014.
- [Li *et al.*, 2019] Shiyang Li, Xiaoyong Jin, Yao Xuan, Xiyu Zhou, Wenhui Chen, Yu-Xiang Wang, and Xifeng Yan. Enhancing the Locality and Breaking the Memory Bottleneck of Transformer on Time Series Forecasting. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.
- [Li *et al.*, 2020] Jiao Li, Yang Liu, Tao Chen, Zhen Xiao, Zhenjiang Li, and Jianping Wang. Adversarial Attacks and Defenses on Cyber-Physical Systems: A Survey. *IEEE Internet of Things Journal*, 7(6):5103–5115, June 2020.
- [Lippe *et al.*, 2022a] Phillip Lippe, Sara Magliacane, Sindy Löwe, Yuki M. Asano, Taco Cohen, and Efstratios Gavves. iCITRIS: Causal Representation Learning for Instantaneous Temporal Effects, June 2022.
- [Lippe *et al.*, 2022b] Phillip Lippe, Sara Magliacane, Sindy Löwe, Yuki M. Asano, Taco Cohen, and Stratis Gavves. CITRIS: Causal Identifiability from Temporal Intervened Sequences. In *Proceedings of the 39th International Conference on Machine Learning*, pages 13557–13603. PMLR, June 2022.
- [Luo *et al.*, 2021] Yuan Luo, Ya Xiao, Long Cheng, Guojun Peng, and Danfeng (Daphne) Yao. Deep Learning-based Anomaly Detection in Cyber-physical Systems: Progress and Opportunities. *ACM Computing Surveys*, 54(5):106:1–106:36, May 2021.
- [Meng *et al.*, 2020] Chao Meng, Xue Song Jiang, Xiu Mei Wei, and Tao Wei. A Time Convolutional Network Based Outlier Detection for Multidimensional Time Series in Cyber-Physical-Social Systems. *IEEE Access*, 8:74933–74942, 2020.
- [Niggemann *et al.*, 2023] Oliver Niggemann, Bernd Zimmering, Henrik Steude, Jan Lukas Augustin, Alexander Windmann, and Samim Multaheb. Machine Learning for Cyber-Physical Systems. In Birgit Vogel-Heuser and Manuel Wimmer, editors, *Digital Transformation: Core Technologies and Emerging Topics from a Computer Science Perspective*, pages 415–446. Springer, Berlin, Heidelberg, 2023.
- [Schölkopf *et al.*, 2021] Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Toward Causal Representation Learning. *Proceedings of the IEEE*, 109(5):612–634, May 2021.
- [Shorten and Khoshgoftaar, 2019] Connor Shorten and Taghi M. Khoshgoftaar. A survey on Image Data Augmentation for Deep Learning. *Journal of Big Data*, 6(1):60, July 2019.
- [Tsipras *et al.*, 2019] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness May Be at Odds with Accuracy. In *International Conference on Learning Representations*, 2019.
- [Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.
- [Wen *et al.*, 2021] Qingsong Wen, Liang Sun, Fan Yang, Xiaomin Song, Jingkun Gao, Xue Wang, and Huan Xu. Time Series Data Augmentation for Deep Learning: A Survey. In *Twenty-Ninth International Joint Conference on Artificial Intelligence*, volume 5, pages 4653–4660, August 2021.
- [Wu *et al.*, 2020] Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, Xiaojun Chang, and Chengqi Zhang. Connecting the Dots: Multivariate Time Series Forecasting with Graph Neural Networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD ’20*, pages 753–763, New York, NY, USA, August 2020. Association for Computing Machinery.
- [Yin *et al.*, 2021] Yuyu Yin, Haoran Xu, Tingting Liang, Manman Chen, Honghao Gao, and Antonella Longo. Leveraging Data Augmentation for Service QoS Prediction in Cyber-physical Systems. *ACM Transactions on Internet Technology*, 21(2):35:1–35:25, March 2021.
- [Zerveas *et al.*, 2021] George Zerveas, Srideepika Jayaraman, Dhaval Patel, Anuradha Bhamidipaty, and Carsten Eickhoff. A Transformer-based Framework for Multivariate Time Series Representation Learning. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, KDD ’21*, pages 2114–2124, New York, NY, USA, August 2021. Association for Computing Machinery.
- [Zhao *et al.*, 2023] Tong Zhao, Wei Jin, Yozen Liu, Yingheng Wang, Gang Liu, Stephan Günnemann, Neil Shah, and Meng Jiang. Graph Data Augmentation for Graph Machine Learning: A Survey, January 2023.
- [Zimmering *et al.*, 2021] Bernd Zimmering, Oliver Niggemann, Constanze Hasterok, Erik Pfannstiel, Dario Ramming, and Julius Pfrommer. Generating Artificial Sensor Data for the Comparison of Unsupervised Machine Learning Methods. *Sensors*, 21(7):2397, January 2021.