ONLINE LEARNING FOR SURROGATE MODEL MAINTENANCE

Balázs Palotai[1,2*], Gábor Kis[1], Tibor Chován[2] and Ágnes Bárkányi[2]

1 MOL Group Plc., Dombóvári út 28, Budapest, 1117, HUNGARY
2 Department of Process Engineering, University of Pannonia, Egyetem utca 10, Veszprém, 8200, HUNGARY

In the process industry, flowsheet models are commonly used to create digital representations of real processes. While these models provide detailed simulations, they often struggle with computational demands in dynamic operating environments. Surrogate models offer a more efficient alternative, but their level of accuracy must be continuously aligned with actual system behavior. This paper presents an MLOps-aligned online surrogate calibration method that maintains and improves surrogate model accuracy by dynamically incorporating localized operational data. Unlike traditional global surrogates trained on static datasets, the proposed approach adapts to changing conditions while preserving previously learned knowledge, effectively addressing the challenge of catastrophic forgetting. Demonstrated on a heat exchanger network, the method significantly improves prediction accuracy in previously unmodeled operating regimes, enhancing the robustness and reliability of digital twins in industrial applications.

Keywords: surrogate model, online learning, digital twin, model maintenance

1. Introduction

Simulation and optimization are essential in the process industry to achieve operational efficiency and ensure system safety [1]. Traditionally, detailed process simulations are carried out using high-fidelity flowsheet models, which support in-depth analysis, scenario evaluation, performance enhancement and equipment condition monitoring [2]. These models, grounded in physical laws and thermodynamic relationships, serve as reliable tools for data-driven decision-making. However, maintaining their accuracy over time can be challenging, especially when integrated into real-time applications where the underlying physical systems evolve continuously [3].

To overcome the computational complexity of flowsheet simulations, surrogate models—also referred to as reduced-order or metamodels—have emerged as a practical alternative [4]. These models emulate the behavior of complex simulations using simplified mathematical frameworks, thereby enabling faster analysis and optimization. Constructing an effective surrogate model typically requires sampling data from the original high-fidelity flowsheet model. Two common approaches are one-shot sampling and adaptive sampling [5]. One-shot techniques, such as Latin Hypercube Sampling (LHS), generate statistically distributed samples across the design space in a single step [6]. In contrast, adaptive sampling strategies iteratively select

new data points to enhance the precision of a model with fewer evaluations [7].

While both sampling strategies are efficient with regard to initial surrogate development [4], they may not ensure a sufficient level of accuracy during its actual operation. The real system may operate in regions that were underrepresented or completely absent in the initial sampling, especially when the operating range of a derived parameter is unknown beforehand. As a result, surrogates trained offline may underperform in dynamic or previously unseen regimes.

To address this gap, online learning offers a way to refine surrogate models during their implementation by incorporating real-time data. This continuous adaptation enables the model to respond to new operational scenarios and maintain predictive accuracy. This study proposes and evaluates an online surrogate learning framework that iteratively updates the model based on observed process data. The goal is to maintain a high level of accuracy across both well-known and emerging operating conditions, enhancing the robustness and reliability of the surrogate in practical industrial settings.

2. Methodology

In practical applications, surrogate models are used to replace high-fidelity simulations for the purpose of

reducing computational load. In this research context, the role of a surrogate model is to support the calibration of the flowsheet model by estimating calibration factors that align the simulation with the real process [8]. To demonstrate the effectiveness of online learning approaches, this study focuses on retraining the surrogate model. In simple terms, the flowsheet model is treated as a stand-in for the real system, generating data that the surrogate is expected to replicate. It is assumed that the flowsheet model accurately represents the real process at all operating points and the surrogate model learns from this data accordingly.

2.1. Problem statement

LHS is a widely used one-shot technique for surrogate-model training due to its ability to maintain generality as dimensionality increases, while ensuring well-distributed sampling across the input space [9]. This method requires predefined minimum and maximum ranges for each input parameter, typically based on the normal operating window of the real process. However, defining appropriate ranges becomes challenging for derived parameters—such as fouling factors in heat exchangers—when prior knowledge is lacking. As illustrated in *Figure 1*, if the surrogate model is trained on hypothetical parameter ranges that differ significantly from the actual distribution observed while in operation, its predictive accuracy can reduce.

Ideally, surrogate-model training would be based on the true parameter ranges relevant to the system, but this information is often unavailable during offline model development, moreover, only becomes accessible once the model is deployed and real operational data are collected.

2.2. Introduction to online learning

Most conventional machine learning approaches operate under the classical paradigm, where the complete dataset is available prior to training. These models are built on the assumption that the data distribution remains static and the underlying structure does not change over time [10]. When new data become available, the typical solution is to retrain the entire model from scratch, which can be computationally intensive and inefficient.

In contrast, online learning offers a framework for continuous model adaptation in response to a stream of incoming data [11]. This approach processes data incrementally, updating the model one sample at a time, facilitating model training to occur sequentially as new data become available:

$$s_1, s_2, \dots, s_i \quad \Rightarrow \quad h_1, h_2, \dots, h_i \qquad (1),$$

where s_i represents the training sample at time step i and h_i denotes the state of the model after learning from that sample. Each training sample is composed of an input–output pair:

$$s_i = (x_i, y_i), \quad x_i \in R^n, \quad y_i \in R^m \qquad (2),$$
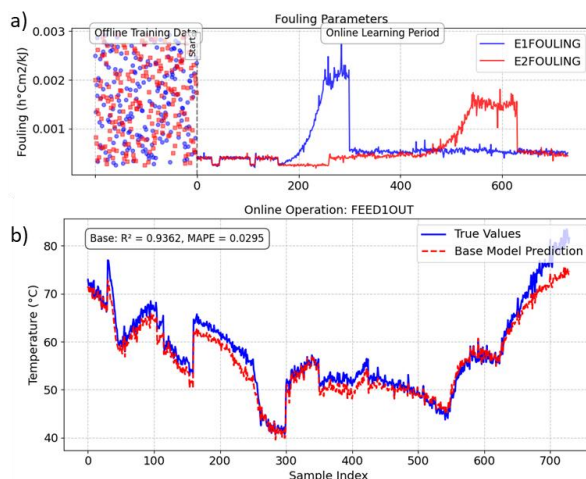

Figure 1: (a) Comparison between the sampling range of the fouling factor (left) and the range of the actual process fouling factor (right); (b) Prediction of the offline-trained base model (red) versus actual process data (blue) from the use case

where x_i stands for the input feature vector with n dimensions and y_i denotes the corresponding labeled output vector with m dimensions. The model at time step i, denoted by h_i, is updated based on the previous model state h_{i-1} and the most recent p samples:

$$h_i = f\big(h_{i-1}, s_i, s_{i-1}, \dots, s_{i-p}\big) \qquad (3).$$

This formulation reflects the online learning paradigm where the model incrementally evolves with each new data point, arbitrarily incorporating a small window of past samples.

Online learning is particularly useful in dynamic environments where data evolve over time and an immediate response is required. This method is commonly employed in systems that operate autonomously, such as in robotics or self-driving vehicles [12], as well as in applications where training signals are provided progressively through human interaction or feedback [13]. Unlike traditional training, which assumes a static environment, online learning enables systems to remain adaptable and responsive to new information as it becomes available.

Online learning offers a range of advantages, particularly in dynamic environments where timely adaptation and efficient data handling are essential. One of its primary strengths lies in efficient resource utilization. Unlike traditional methods that require access to the entire dataset, online learning processes incoming data incrementally, reducing memory and computational demands by only storing and updating a limited subset at any given time [14].

Another key advantage is real-time adaptation. Models can adjust their predictions and internal parameters continuously, allowing them to respond immediately to changes in the environment or system behavior. This feature is especially useful in scenarios where system conditions evolve rapidly, such as in financial forecasting, industrial automation or non-

stationary domains like weather prediction, where patterns shift over time and data relevance quickly reduces [15].

However, online learning also presents several challenges. A notable limitation is catastrophic forgetting, a phenomenon where the model gradually loses previously acquired knowledge while adapting to new data [16]. This can compromise long-term model stability and degrade performance in tasks learned earlier. Furthermore, handling concept drift—sudden or gradual shifts in the underlying data distribution—remains a significant challenge, often requiring advanced strategies to detect and adapt to such changes effectively.

Finally, an inherent risk of overfitting in online settings must be addressed. Since updates are made frequently and often based on small or recent data segments, the model may become overly sensitive to recent fluctuations, leading to poor generalization. Balancing adaptability with robustness is therefore a critical aspect in the design of effective online learning systems [17].

2.3. Compared methodologies

In real-world applications, online learning provides an effective framework for continuously adapting surrogate models to reflect the behavior of the actual process. However, ensuring long-term reliability requires careful design to prevent overfitting to recent data and mitigate catastrophic forgetting, where the model loses its ability to generalize from past knowledge.

To evaluate the effectiveness of different online learning strategies, this study compares several update mechanisms, each with distinct learning dynamics and memory usage:

- Incremental: The model is updated as each new data point arrives (*Figure 2a*), allowing for immediate adaptation but with an increased risk of forgetting prior knowledge.
- Mini-Batch: The model accumulates a small number of new data points, namely 5, before performing an update (*Figure 2b*). This buffering approach smooths out fluctuations and reduces overfitting to individual samples.
- Incremental Replay: This is similar to the incremental approach, but a small proportion (1%) of the original offline training data is blended into each new data point (*Figure 2c*) in order to preserve past knowledge while adapting to new patterns.
- Cumulative: With each update, the model is trained using the complete offline dataset combined with all new incoming data (*Figure 2d*). This strategy provides strong memory retention but increases computational cost over time.
- Hybrid: The model updates incrementally with each new data point. However, when prediction accuracy drops below a practical threshold, the corresponding data points are stored and later used in a cumulative update after a defined number of samples, in this case 5, is collected (*Figure 2e*). The aim of this approach is to balance adaptability against long-term
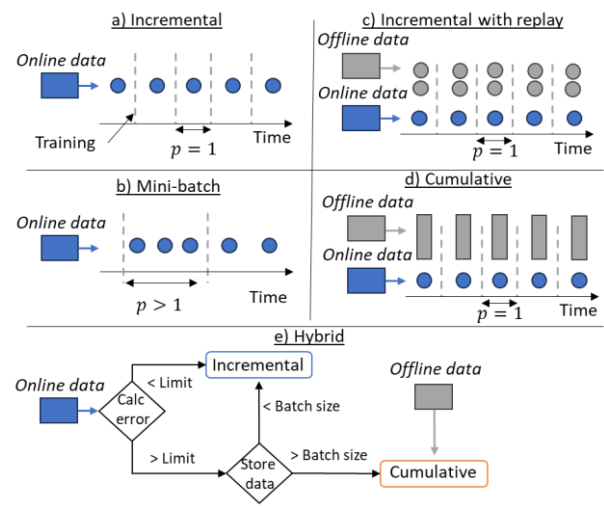

Figure 2: Illustration of compared online learning methodologies.

stability by selectively reinforcing learning as the level of performance declines. The practical threshold reflects the practical requirements of implementing the surrogate model. It is important to recognize that pursuing the highest possible degree of accuracy does not always translate into improved practical outcomes and may, in fact, increase the risk of overfitting.

2.4. Metrics

The primary objective of continuously retraining surrogate models is to maintain or improve predictive accuracy as new operational data become available. Ideally, each successive version of a model should outperform its predecessor at the next working point. To evaluate this progression in real time, a modified version of the Mean Absolute Scaled Error (MASE) is introduced.

Unlike the traditional MASE metric—which evaluates forecast accuracy by comparing the performance of a model to that of a naïve baseline [18]—the modified MASE proposed here compares the error of the newly updated model to that of the previously deployed model on the same data point. This adaptation provides a direct measure of improvement in performance between sequential updates to a model.

$$Modified\ MASE_t = \frac{MAE_t^{(p)}}{MAE_t^{(c)}} \quad (4),$$

$$MAE_t^{(c)} = \frac{1}{n}\sum_{i=1}^{n}\left|y_t^{(i)} - \hat{y}_t^{(c,i)}\right| \quad (5),$$

$$MAE_t^{(p)} = \frac{1}{n}\sum_{i=1}^{n}\left|y_t^{(i)} - \hat{y}_t^{(p,i)}\right| \quad (6),$$

where:
- t: the current time step or working point,
- n: the number of output variables,

Table 1: Use case model input - output parameters

Type	Description	Uom
Input	FEED1 mass flow rate	kg/h
Input	FEED1 temperature	°C
Input	PROD1 mass flow rate	kg/h
Input	PROD1 temperature	°C
Input	E1 Fouling factor	h°Cm²/kJ
Input	E2 Fouling factor	h°Cm²/kJ
Output	PROD1E1IN temperature	°C
Output	FEED1E1OUT temperature	°C
Output	FEED1OUT temperature	°C
Output	PROD1OUT temperature	°C

- $y_t^{(i)}$: the true value of output i at time t
- $\hat{y}_t^{(c,i)}$: the prediction of the current model after online retraining,
- $\hat{y}_t^{(p,i)}$: the prediction of the previously deployed model before an update.

However, this metric only reflects the relative level fo improvement between versions of a model in terms of the latest sample and does not account for potential degradation in the overall accuracy of the model, particularly in terms of the original training distribution. To address this, each updated model is also evaluated with regard to both the original offline and online datasets. This dual evaluation ensures that gains in local adaptability do not come about at the expense of broader model robustness.

3. Experimental

3.1. Introduction to the Use Case

The proposed online learning strategies were evaluated using a simplified heat exchanger network model. The flowsheet model was developed using Aspen HYSYS V14 (*Figure 3*), moreover, the corresponding input and output parameters used for surrogate modeling are summarized in *Table 1*.

To implement the surrogate model, the neural network architecture was chosen due to its flexibility in terms of capturing complex nonlinear relationships. Specifically, the MLPRegressor from scikit-learn was used as it provides native support for incremental learning via the partial_fit() method that allows the model to be updated progressively with new data without reinitializing the learned weights, which is essential in online learning scenarios.

The initial surrogate model was constructed using LHS, generating 200 samples from the flowsheet model. A feedforward neural network with one hidden layer, containing 50 neurons, was trained on this dataset. The input sampling range was predefined based on expected operational bounds for the system, serving as the baseline
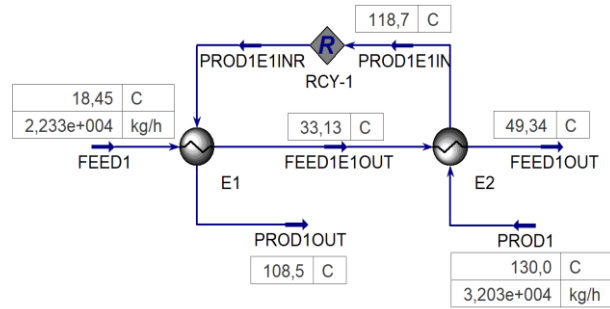

Figure 3: Flowsheet representation of the heat exchanger network modelled in Aspen HYSYS V14 in the use-case scenario.
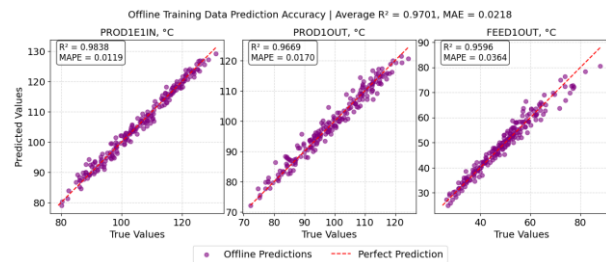

Figure 4: Scatter plot illustrating the predictive accuracy of the base surrogate model in terms of the monitored output parameters.
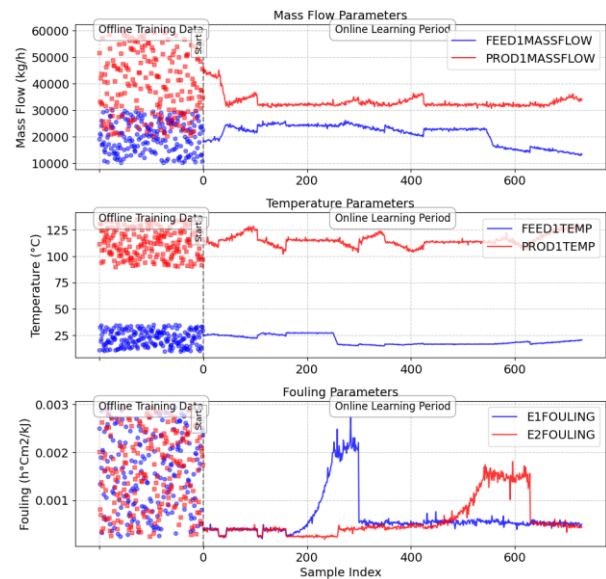

Figure 5: Visualization of the generated test "online" dataset, showing offline data points on the left and sequentially introduced online data on the right.

for both surrogate training and subsequent online updates. The baseline model achieved an average R² of 0.9701 and a Mean Absolute Percentage Error (MAPE) of 0.0218 across the output parameters (*Figure 4*).

3.2. Test "online" data

To evaluate the performance of the online learning strategies, a test dataset consisting of 730 data points was

generated. This dataset simulates realistic process variability by introducing gradual changes to input parameters, added noise and the effects of heat exchanger fouling (*Figure 5*). In contrast, an ideal scenario—where the test data is available during offline training and combined with the LHS dataset—would allow the model to achieve an enhanced level of performance with an average R² of 0.988 and a MAPE of 0.0069 (as shown in *Figure 6*). This comparison highlights the potential performance gap that online learning aims to bridge by progressively incorporating new data into the model.

3.3. Online learning workflow

The performance of the base surrogate model was evaluated using the test dataset in a sequential manner, simulating a real-time operation. New data points were introduced one by one, allowing the model to be assessed under conditions that reflect actual operational dynamics. At each working point, the predictive accuracy of the model was recorded both in terms of the new and original dataset to evaluate its ability to adapt to changing conditions and robustness with regard to retaining previously learned knowledge. Based on the selected online learning strategy, the model was retrained continuously using the incoming data, following the specific update procedures described in Section 2.3.

4. Results and discussion

The results are evaluated from two perspectives: adaptability, which reflects how effectively the model adjusts to new operating conditions, and robustness, which measures the model's ability to maintain its level of performance in terms of the original offline LHS dataset.

4.1. Adaptability

The adaptability performance of all the evaluated online learning strategies assessed using the modified MASE metric is presented in *Figure 7*. In this context, a MASE value greater than 1 indicates that the updated model performed better than its previous version at a given working point with higher values denoting an improved level of adaptability to new data.

All the evaluated strategies demonstrated adaptability, consistently achieving MASE values above 1. This confirms that each method was able to effectively learn from incoming data points and improve model predictions over time.

Among the tested approaches, the hybrid and incremental learning strategies exhibited the strongest level of adaptability, consistently outperforming the other methods throughout the dataset. This outcome is in line with expectations as Mini-batch learning delays updates to models until multiple data points, e.g. 5, are collected, which slows the level of responsiveness to sudden changes. Cumulative learning, while comprehensive, tends to underrepresent the newest data
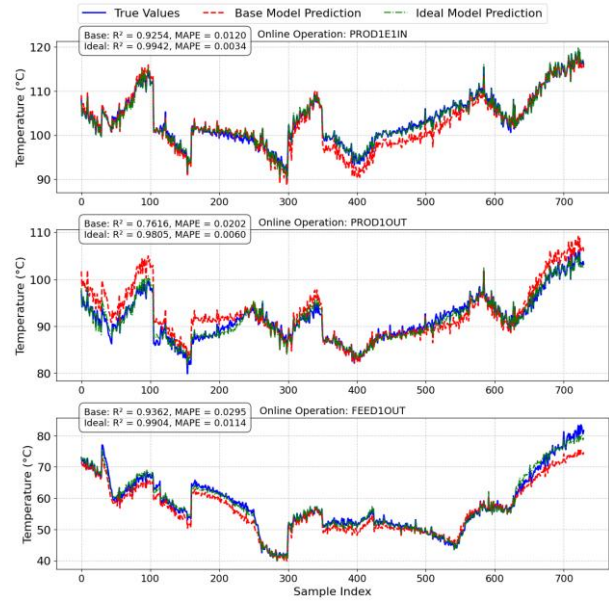

Figure 6: Comparison between the base surrogate model and ideal model prediction accuracy of the test dataset
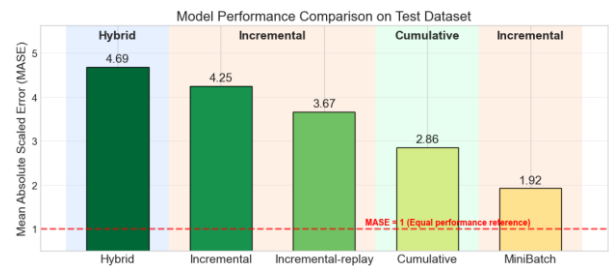

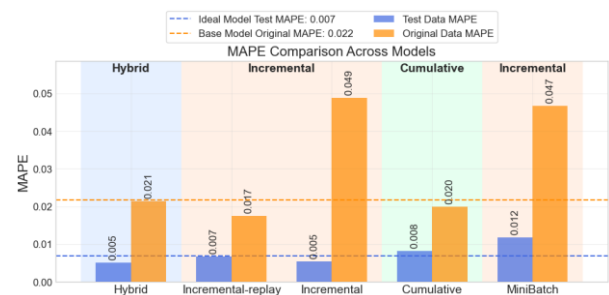*Figure 7:* Adaptability performance of online learning solutions, measured using the modified MASE metric

Figure 8: Robustness performance of online learning solutions with regard to the original LHS data (orange) and the level of adaptability to the test "online" data (blue)

during retraining due to the overwhelming influence of the older, larger offline dataset.

4.2. Robustness

Figure 8 compares the performance of the various online learning strategies against both the original surrogate model (baseline) and an ideal offline-trained model. This

comparison is conducted across two datasets: the test dataset—representing evolving operational conditions—and the original LHS dataset used for initial model training. The goal is to approach the performance of the ideal model with regard to the test data while preserving the accuracy of the baseline model in terms of the original data, thereby achieving a balance between adaptability and robustness.

The hybrid learning approach emerged as one of the most effective strategies. It achieved a high degree of accuracy in terms of the test data, comparable to the ideal offline model, while also preserving the original performance of the surrogate model concerning the LHS dataset. This confirms its ability to maintain previously acquired knowledge while adapting effectively to new conditions.

The incremental learning strategy also performed strongly with regard to the test dataset, reflecting a high level of adaptability. However, it suffered a noticeable drop in accuracy in terms of the original LHS dataset, suggesting signs of catastrophic forgetting. This issue was partially mitigated by incorporating a 1% replay of the original training data during each update cycle. While this replay strategy helped preserve the level of performance in terms of the original dataset, it introduced a slight reduction in predictive accuracy with regard to the test data—highlighting the trade-off between retaining old knowledge and adapting to new trends.

On the other hand, the cumulative learning strategy prioritized retention of the original training data and achieved similar levels of accuracy in terms of the LHS dataset. However, this came at the cost of a lower level of performance regarding the test dataset as the large volume of historical data overshadowed the influence of the newly acquired samples, thereby reducing its ability to adapt swiftly to evolving process conditions.

The Mini-Batch learning method, while showing a modest degree of improvement compared to the baseline on the test dataset, yielded the lowest level of accuracy during the test among the tested approaches. Additionally, infrequent updates to this model were insufficient to maintain robustness in terms of the original dataset, resulting in a suboptimal level of performance with regard to both adaptability and robustness.

In summary, the hybrid learning method demonstrated the most effective trade-off between adaptability and robustness, achieving a strong level of accuracy in terms of the test dataset as shown in *Figure 9* and still a good degree of accuracy with regard to the original dataset shown in *Figure 8*. Incremental learning offered a strong level of adaptability but required mitigation strategies to preserve older knowledge. Cumulative learning ensured robustness but limited adaptability, while Mini-Batch learning lagged behind due to its delayed response to new data.
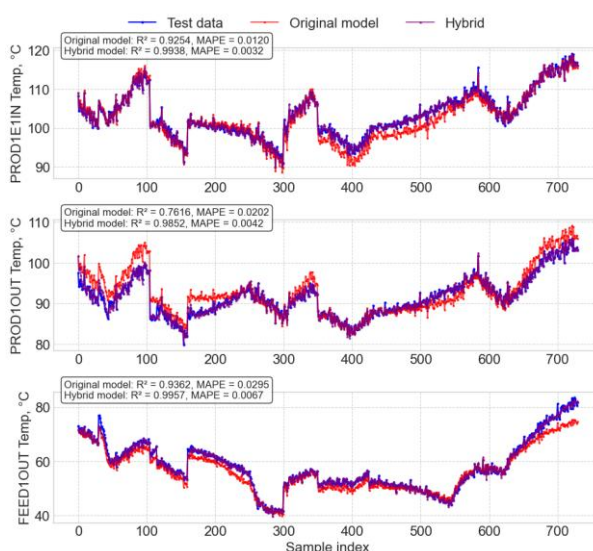

Figure 9: Comparison of the accuracy of the base model and hybrid learning model with regard to the online test data

5. Conclusions

The proposed online learning approaches demonstrate significant potential in terms of enhancing the adaptability of surrogate models under evolving operating conditions. All the investigated methods contributed to refining the base surrogate model by enabling effective adaptation to new data. A critical challenge in online learning is the preservation of previously acquired knowledge, which was successfully addressed by the hybrid approach, combining the strengths of both incremental and cumulative learning strategies. Additionally, in terms of incremental learning, a small replay, 1% in this use case, of the original training data is incorporated during each update cycle, ensuring that essential past information is retained while allowing the model to adapt to new data.

Achieving an optimal trade-off between adaptability and robustness requires careful calibration of the online learning process. In particular, setting a realistic accuracy threshold is essential, namely one that reflects the practical requirements necessary to implement the surrogate model. It is important to recognize that pursuing the highest possible level of accuracy does not always translate into improved practical outcomes and may, in fact, increase the risk of overfitting. Therefore, the design of online learning strategies should consider both the enhancement of performance and long-term stability of the model in order to be reliably deployed in real-world industrial systems. Additionally, ongoing monitoring of model performance is essential to maintain its credibility and ensure it continues to meet operational requirements.

Acknowledgements

REFERENCES

[1] Tarcsay, B.L.; Bárkányi, Á.; Chován, T.; Németh, S.: Development of compartment models for diagnostic purposes, *Hung. J. Ind. Chem.*, 2021, **49**(1), 47–58, DOI: 10.33927/hjic-2021-07

[2] Dimian, A.C.; Bildea, C.S.; Kiss, A.A.: Chapter 2 - Introduction in process simulation, in: Computer aided chemical engineering: Integrated design and simulation of chemical processes, Dimian, A.C.; Bildea, C.S.; Kiss, A.A. (Eds.) (Elsevier), 2014, **35**, pp. 35–71, DOI: 10.1016/B978-0-444-62700-1.00002-4

[3] Pan, I.; Mason, L.R.; Matar, O.K.: Data-centric engineering: integrating simulation, machine learning and statistics. Challenges and opportunities, *Chem. Eng. Sci.,* 2022, **249**, 117271, DOI: 10.1016/j.ces.2021.117271

[4] Bárkányi, Á.; Chován, T.; Németh, S.; Abonyi, J.: Modelling for digital twins—potential role of surrogate models, *Processes*, 2021, **9**(3), 476, DOI: 10.3390/pr9030476

[5] Bhosekar, A.; Ierapetritou, M.: Advances in surrogate based modeling, feasibility analysis, and optimization: A review, *Comput. Chem. Eng.,* 2018, **108**, 250–267, DOI: 10.1016/j.compchemeng.2017.09.017

[6] Hadigol, M.; Doostan, A.: Least squares polynomial chaos expansion: A review of sampling strategies, *Comput. Methods Appl. Mech. Eng.,* 2018, **332,** 382–407, DOI: 10.1016/j.cma.2017.12.019

[7] Dias, L.; Bhosekar, A.; Ierapetritou, M.: Adaptive sampling approaches for surrogate-based optimization, in: Computer aided chemical engineering: Proceedings of the 9th International Conference on Foundations of Computer-Aided Process Design, Muñoz, S.G.; Laird, C.D.; Realff, M.J. (Eds.) (Elsevier), 2019, **47**, 377–384, DOI: 10.1016/B978-0-12-818597-1.50060-6

[8] Palotai, B.; Kis, G.; Abonyi, J.; Bárkányi, Á.: Surrogate-based flowsheet model maintenance for Digital Twins, *Digit. Chem. Eng.*, 2025, **15**, 100228, DOI: 10.1016/j.dche.2025.100228

[9] Hao, Z.; Zhang, C.; Lapkin, A.A.: Efficient surrogates construction of chemical processes: Case studies on pressure swing adsorption and gas-to-liquids, *AIChE J.*, 2022, **68**(6), e17616, DOI: 10.1002/aic.17616

[10] Gepperth, A.; Hammer, B.: Incremental learning algorithms and applications, *European Symposium on Artificial Neural Networks (ESANN)*, 2016, Bruges, Belgium, hal-01418129 https://hal.science/hal-01418129v1 (retrieved 25 April 2025)

[11] Montiel, J.; Halford, M.; Mastelini, S.M.; Bolmier, G.; Sourty, R.; Vaysse, R.; Zouitine, A.; Gomes, H.M.; Read, J.; Abdessalem, T.; Bifet, A.: River: machine learning for streaming data in Python, *J. Mach. Learn. Res.,* 2021, **22**(110)**,** 1–8

[12] Hammer, B.; Toussaint, M.: Special issue on autonomous learning, *KI – Künstl. Intell.*, 2015, **29**(4), 323–327, DOI: 10.1007/s13218-015-0392-x

[13] Honeycutt, D.; Nourani, M.; Ragan, E.: Soliciting human-in-the-loop user feedback for interactive machine learning reduces user trust and impressions of model accuracy, *Proc. AAAI Conf. Hum. Comput.,* 2020, **8**(1), 63–72, DOI: 10.1609/hcomp.v8i1.7464

[14] Hoi, S.C.H.; Sahoo, D.; Lu, J.; Zhao, P.: Online learning: A comprehensive survey, *Neurocomputing*, 2021, **459**, 249–289, DOI: 10.1016/j.neucom.2021.04.112

[15] Losing, V.; Hammer, B.; Wersing, H.: Incremental on-line learning: A review and comparison of state of the art algorithms, *Neurocomputing,* 2018, **275**, 1261–1274, DOI: 10.1016/j.neucom.2017.06.084

[16] Yang, R.; Yang, T.; Yan, Z.; Krajnik, T.; Ruichek, Y.: Preventing catastrophic forgetting in continuous online learning for autonomous driving, *2024 IEEE/RSJ Int. Conf. Intell. Rob. Sys. (IROS), Abu Dhabi, United Arab Emirates,* 2024, 5505–5512, DOI: 10.1109/IROS58592.2024.10801619

[17] Ying, X.: An overview of overfitting and its solutions, *J. Phys. Conf. Ser.*, 2019, **1168**(2), 022022, DOI: 10.1088/1742-6596/1168/2/022022

[18] Hyndman, R.J.; Koehler, A.B.: Another look at measures of forecast accuracy, *Int. J. Forecast.*, 2006, **22**(4), 679–688, DOI: 10.1016/j.ijforecast.2006.03.001