Separation and Purification Technology 397 (2026) 137951

Contents lists available at ScienceDirect

Separation and Purification Technology

journal homepage: www.elsevier.com/locate/seppur

Dynamic grey box modeling of decanter centrifuges via continual learning☆

Ouwen Zhai *, Helene Rumiko Reppich, Hermann Nirschl, Marco Gleiss

Institute of Mechanical Process Engineering and Mechanics, Karlsruhe Institute of Technology, Karlsruhe, 76131, Germany

ARTICLE INFO

Communicated by B. Van der Bruggen

Keywords:
Continual learning
Decanter centrifuge
Machine learning
Grey box modeling
Neural network

ABSTRACT

Decanter centrifuges are essential for solid–liquid separation across multiple industrial sectors, yet their operation still depends heavily on expert knowledge and time-intensive experimentation. Existing predictive models, including white box, black box and hybrid grey box models, typically require static offline calibration on empirical data. This limits their ability to adapt to changing material properties during operation, which leads to diminished prediction accuracy under dynamic operating conditions.

This study addresses this limitation by integrating a Continual Learning (CL) framework into a grey box model for decanter centrifuge operation, enabling the model to adapt online to changing material properties and process behavior. While CL has shown promise in machine learning, its application in process and chemical engineering remains limited. To mitigate catastrophic forgetting, defined as the gradual loss of previously acquired knowledge during sequential learning, the Online Elastic Weight Consolidation (EWC) method is employed. The proposed approach is evaluated through an application-driven case study of a dynamic mill-decanter process circuit.

Results demonstrate that the proposed model adapts reliably online while maintaining memory stability and generalization, outperforming a statically calibrated baseline under changing operating conditions. This adaptability enhances operational stability and enables more responsive process control, highlighting strong potential for real-time deployment in complex and dynamic industrial processes.

1. Introduction

Decanter centrifuges are continuously operating solid-bowl centrifuges extensively employed in the chemical, food and mineral processing industries, where they are central for solid–liquid separation and classification. Their efficient operation is essential for energy- and resource-efficient production, making reliable predictive models increasingly important for sustainable process design and control. Nonetheless, decanter centrifuges are still operated largely based on expert knowledge, and achieving the desired separation performance often requires extensive experimental testing.

To optimize the operation and reduce experimental effort, a variety of predictive models have been developed. These can be broadly categorized into physics-based white box models (WBMs), data-driven black box models (BBMs), and hybrid grey box models (GBMs). WBMs include both fully resolved CFD simulations and compartment-based models. For instance, Zhu et al. [1] used an Euler–Euler three-phase CFD approach to analyze the flow behavior and separation performance in a three-phase decanter centrifuge for oil-water-solid separation. Baust et al. [2] presented an OpenFOAM solver that incorporates empirical material functions to simulate and optimize the clarification and classification performance in decanter centrifuges. Although CFD approaches allow accurate simulation of decanter centrifuges, their computational cost is substantial, limiting use in real-time applications.

In contrast, real-time compartment-based models reduce system complexity with simplifying assumptions, drastically reducing computational demand. Bai et al. [3] proposed such an approach to evaluate separation and classification performance for several flow profiles, including plug flow, parabolic flow, and a profile that incorporates backflow near both the bowl wall and the sediment surface. However, their model does not include the consolidation behavior of compressible solids. Gleiss et al. [4] and Menesklou et al. [5] developed a comparable approach using empirical functions to describe hindered settling and sediment compaction. Nevertheless, these simplified models depend heavily on empirical parameters which may vary significantly with changing feed properties.

To address these limitations, Zhai et al. [6] introduced a GBM in which a neural network (NN) predicts correction parameters for the empirical functions, which are then fed into the compartment model, significantly improving prediction accuracy. Yet, this model still requires offline training on static datasets, and its predictive performance

☆ This article is part of a Special issue entitled: 'WFC14' published in Separation and Purification Technology.

* Corresponding author.
E-mail address: Ouwen.Zhai@kit.edu (O. Zhai).

https://doi.org/10.1016/j.seppur.2026.137951
Received 27 November 2025; Received in revised form 18 March 2026; Accepted 10 April 2026
Available online 22 April 2026

degrades when material properties drift during operation. This limits its applicability in dynamic process environments, which are often found in wastewater treatment, mineral slurry dewatering, and food processing, where material properties are highly variable. Continual Learning (CL), also referred to as Incremental Learning and Lifelong Learning, addresses exactly this problem by enabling models to update incrementally as new data become available [7,8]. Unlike naive retraining on static datasets, which is computationally expensive, CL methods fine-tune model parameters instead of completely redefining them. A central challenge in CL is the gradual loss of previously acquired knowledge when learning new data, also known as catastrophic forgetting. Numerous methods have been developed to mitigate catastrophic forgetting [9–14].

While CL has been studied extensively in vision and language domains, there is a significant gap of CL across other domains [15]. Even though already used in the field of medicine [16,17], or in the field of robotics [18], only a limited number of works explore its application to process or chemical engineering systems. Tercan et al. [19] used transfer learning and CL techniques to improve predictive accuracy of a NN model for injection molding. Maschler et al. [20] proposes a CL approach for industrial predictive maintenance. Their method is evaluated on a turbofan engine fault prediction case study, where it demonstrates the ability to sequentially learn from small sub-datasets while mitigating forgetting of previously learned tasks. Nevertheless, the performance decreases when the tasks vary significantly in complexity. Fu et al. [21] integrated CL into a physics-guided model for redox-flow batteries. However, studies involving realistic industrial processes, and in particular separation processes, remain scarce.

In this work, we address this gap by integrating a CL framework into a GBM for decanter centrifuges, enabling continuous online adaptation to evolving material properties, while preserving previously learned knowledge. The proposed framework enhances predictive robustness in dynamic industrial environments compared to current models and represents a significant advance toward sustainable, real-time model-based control of separation processes, while further contributing domain-specific insights to an area where such methods remain underexplored.

2. Background and related work

2.1. Decanter centrifuge model

To model the separation process in decanter centrifuges, we employ a grey box model (GBM) described in the work of Zhai et al. [6]. The underlying white box model (WBM), developed by Gleiss et al. [4] and Menesklou et al. [5], represents sedimentation, sediment dewatering and sediment transport using compartment-based discretization along the helical screw conveyor. Each compartment contains suspension and sediment regions governed by mass balances and empirical material functions, including a hindered settling and a sediment compaction function. However, limitations in the WBM arise from uncertainties in sediment compaction and settling behavior. Zhai et al. [6] address these limitations by introducing a GBM with a serial structure. Given input parameters, including the feed volumetric flow rate $\dot{V}_{\mathrm{in}}$, feed solids mass fraction ϕ_{in}, rotational speed n_{rot}, differential speed Δn_{rot}, pool depth h_{pool}, and the characteristic particle sizes $x_{10,3}$, $x_{50,3}$ and $x_{90,3}$, a multi-layer perceptron (MLP) predicts two physically meaningful intermediate parameters, namely a sediment compaction parameter $\phi_{\mathrm{v,sh}}$ and a correction factor for hindered settling H_{corr}. Because these parameters cannot be measured directly, they are identified through Bayesian optimization. The intermediate material-related parameters, additional material data, process data, and machine geometry serve as input of the white box model, which outputs the solids mass fraction and particle size distribution (PSD) of the centrate and sediment. Fig. 1 shows the model arrangement, input parameters, and output parameters. The architecture of the MLP and its hyperparameters are summarized in Table A.1.

The resulting GBM achieves good predictive accuracy and extrapolation capability. However, it still relies on labor-intensive material characterization and is sensitive to variations in material properties. Deviations in feed characteristics, such as those caused by upstream comminution can therefore degrade model performance. Restoring model accuracy would require computationally expensive offline retraining. To overcome this limitation, we extend the GBM with a CL framework that enables incremental online adaptation to changing material properties while avoiding complete retraining.

2.2. Continual Learning

Continual Learning (CL) is defined as the ability of machine learning models to continuously acquire and adapt knowledge without forgetting previously learned information. The simplest baseline method is the naive fine-tuning of existing models. Although it can lead to successful learning of the new task, this method is prone to catastrophic forgetting. A large variety of methods are available to mitigate catastrophic forgetting. Generally, they can be categorized into regularization-based methods, architectural methods, and replay-based methods.[7,15]

Replay-based methods involve training the model on new datasets while simultaneously replaying previously learned data points. The old samples are stored within memory buffers. [11,12,22,23] Although predictive performance and memory stability can be strong, replay-based methods are limited by the need for sufficiently large pre-trained datasets when large quantities of new data are introduced. Moreover, since the old data must be stored, memory requirements increase with each incremental training step.

Architectural methods follow a different strategy, where task-specific parameters are constructed [13,24–26]. For example, Rusu et al. [24] introduce progressive networks which add subnetworks to an existing structure specifically for new tasks while keeping the old subnetwork frozen. While effective in mitigating catastrophic forgetting, the network size scales with each additional task, increasing computational time.

Parameter regularization methods add a regularization term to the cost function that selectively penalizes the variation of previously important weights [9,10,14,27]. For example, the Synaptic Intelligence (SI) method, proposed by Zenke et al. [14], represents an approach where the importance of each weight is determined and recorded during every iteration of backpropagation. Kirkpatrick et al. [9] propose a similar approach with the Elastic Weight Consolidation (EWC) method, by assessing the importance of individual weights with the Fisher matrix to mitigate catastrophic forgetting. The EWC method uses a Bayesian view to limit catastrophic forgetting by penalizing changes to parameters that were important in previous tasks. Let θ be the model parameters and let $T_{1:Z}$ denote tasks 1 to Z. The posterior over parameters after observing all Z tasks can be expressed recursively as:

$$p(\theta \mid T_{1:Z}) \propto p(\theta \mid T_{1:Z-1}) p(T_Z \mid \theta) \tag{1}$$

Since the exact posteriors are intractable for NNs, the EWC method applies a Laplace approximation to the previous posterior around its maximum a posteriori solution (MAP) $\tilde{\theta}$ approximating it as a Gaussian [28] with precision given by the diagonal of the Fisher matrix F. The Fisher matrix itself describes the importance of each weight whereas a weight that is critical for the original dataset has a high value. Using this approximation gives the EWC loss term for the current Task T_Z [9]:

$$L(\theta) = \underbrace{L_Z(\theta)}_{\text{cost function}} + \underbrace{\frac{\lambda_{\mathrm{EWC}}}{2} \sum_{j=1}^{Z-1} \sum_{i=1}^{I} F_j^{(i)} \cdot \left(\theta^{(i)} - \tilde{\theta}_j^{(i)}\right)^2}_{\text{EWC term}} \tag{2}$$

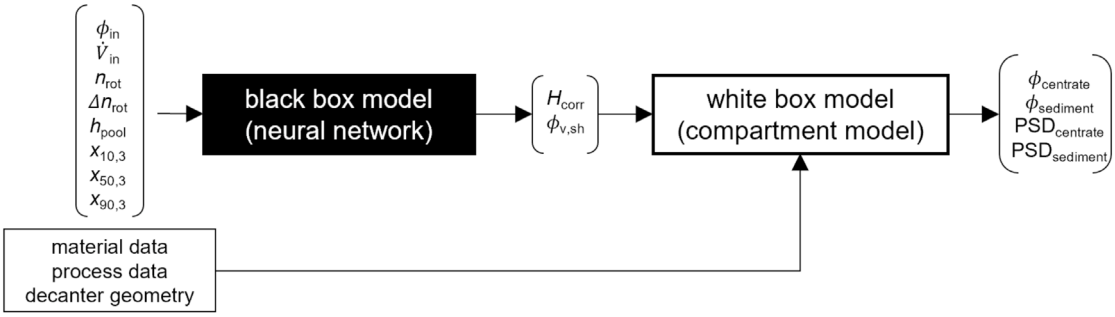

Fig. 1. Structure of the grey box model consisting of a neural network (black box model) and a compartment-based model (white box model).

where $L_Z(\theta)$ is the task loss for the current task, i labels each weight, j labels the previous tasks, I is the total number of weights, $\tilde{\theta}_j$ is the MAP, and λ_{EWC} determines the regularization strength. Minimizing the EWC loss function discourages large changes of previously important weights. One weakness of the EWC algorithm is that it struggles with a large number of incremental tasks due to the linear growth of the regularization term. Schwarz et al. [10] modified the EWC method by consolidating all Fisher matrices into a single Fisher matrix F_Z^* instead of storing one for each task, calling the method online EWC. This leads to the following loss function for task Z:

$$L(\theta) = \underbrace{L_Z(\theta)}_{\text{cost function}} + \underbrace{\frac{\lambda_{\text{EWC}}}{2} \sum_{i=1}^{I} F_{Z-1}^{*(i)} \cdot \left(\theta^{(i)} - \tilde{\theta}_{Z-1}^{(i)}\right)^2}_{\text{online EWC term}} \tag{3}$$

with the Fisher matrix updated after task T_Z as [10]:

$$F_Z^* = \gamma \cdot F_{Z-1}^* + F_Z \text{ with } \gamma \in [0, 1] \tag{4}$$

F_Z is the Fisher matrix for task T_Z, and γ is a decay parameter, controlling how quick older information is discounted. This approach treats all tasks equivalently and avoids the need for task labels. Furthermore, the control parameter γ enables further control of balancing the importance of new and old tasks. We use the online EWC method in combination with the MSE cost function in our framework as it is light on memory, easy to tune, has low computation time and tasks labels are not required.

Wang et al. [7] and Yang et al. [15] provide a more comprehensive and detailed overview of current CL-methods and developments regarding CL.

3. Continual learning framework

The proposed Continual Learning (CL) workflow is summarized in Fig. 2. First, the pre-trained neural network (NN) is loaded and initialized together with all constants and internal buffers. At each time step, the current process and feed conditions are read, and the correction parameters are predicted by the NN. These predicted parameters, together with the process inputs, are subsequently passed to the compartment model to simulate the decanter operation. Novel data points are identified by a novelty-detection module that combines a reconstruction-error criterion and a probability-density-based criterion.

The reconstruction-error-based criterion δ compares the model prediction ϕ_{pred} with experimentally measured solids mass fractions ϕ_{exp} with a defined threshold δ_{lim} and is defined as:

$$\delta = \frac{\left|\phi_{\text{pred}} - \phi_{\text{exp}}\right|}{\phi_{\text{exp}}} \tag{5}$$

The probability-density-based criterion evaluates the likelihood of the observed particle size distribution (PSD) under a Gaussian reference distribution constructed from the most recently observed PSD. If the likelihood falls below a predefined threshold, the sample is classified as

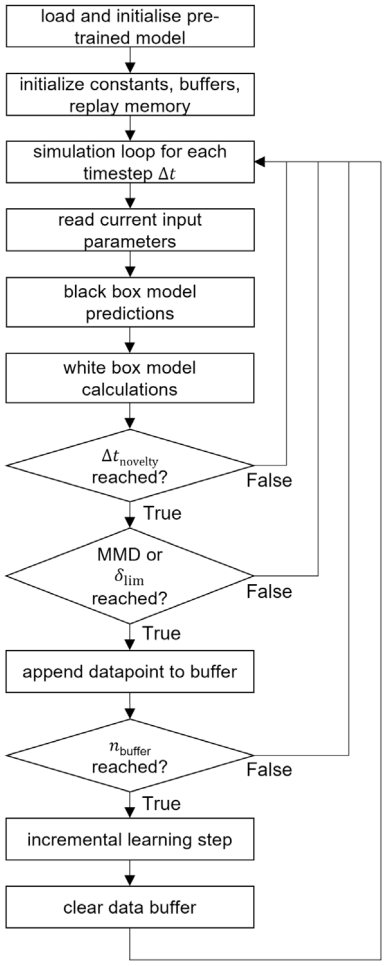

Fig. 2. Workflow of the proposed Continual Learning framework. $\Delta t_{\text{novelty}}$ is the time interval for novelty-check, MMD is the maximum mean discrepancy (see Eq. (6)), δ_{lim} the deviation threshold (see Eq. (5)), and n_{buffer} is the buffer size.

novel. Because the correction parameters represent material-dependent sedimentation and sediment compaction behavior, shifts in PSD directly alter the material behavior. PSD-based novelty detection therefore enables early anticipation of material behavior drift before large prediction errors occur. A common measure for detecting shifts in data distributions is the Maximum Mean Discrepancy (MMD), proposed by Gretton et al. [29]. MMD quantifies the difference between two probability distributions. Given the old samples $P = \{p_h\}_{h=1}^{n}$ and new samples $Q = \{q_j\}_{j=1}^{m}$, MMD computes the distance of their mean embeddings

Table 1
Continual Learning framework parameters for the case study. This table summarizes the key parameters used in the online adaptation study.

Parameter		Value	Dimension
n_{buffer}	buffer size	4	–
δ_{lim}	deviation threshold (see Eq. (5))	0.05	–
α	learning rate	1	–
MMD	maximum mean discrepancy (see Eq. (6))	0.003	–
$\Delta t_{\text{novelty}}$	novelty check time interval	30	s
n_{epoch}	number of epochs	100	–
γ	online EWC parameter	0.9	–
–	optimizer	adam	–
Δt	simulation timestep	1	s

based on the chosen kernel K [29]:

$$\text{MMD}^2(P,Q) = \frac{1}{n^2}\sum_{h=1}^{n}\sum_{j=1}^{n}K(p_h,p_j) + \frac{1}{m^2}\sum_{h=1}^{m}\sum_{j=1}^{m}K(q_h,q_j) - \frac{2}{nm}\sum_{h=1}^{n}\sum_{j=1}^{m}K(p_h,q_j) \tag{6}$$

The first term describes the similarity within the old samples, the second term the similarity within the new samples, and the third term the similarity between old and new samples. Using a Gaussian (RBF) kernel [29], MMD is zero if both distributions are identical. This makes MMD an interpretable and computationally efficient measure for novelty detection. In the presented work, the computational effort for calculation of MMD is negligible. A data point is classified as novel if at least one of the two criteria is met. Novelty detection is executed every t_{novelty} seconds to save computational cost.

To ensure robust retraining behavior, novel samples are accumulated in a buffer and incremental training is triggered only once the batch size limit n_{buffer} is reached. Once it is reached, the correction parameters are first re-estimated via Bayesian optimization under the corresponding feed and process conditions. The updated targets are then used for supervised incremental training. Since the parameter $\phi_{\text{v,sh}}$ primarily affects sediment compaction and has negligible influence on the centrate, the parameters H_{corr} and $\phi_{\text{v,sh}}$ are optimized sequentially. This reduces the computational effort compared to joint optimization while maintaining sufficient accuracy. The loss function L_{opt} used for the optimization is defined as the squared error between the final model prediction and the experimental solids mass fraction:

$$L_{\text{opt}} = (\phi_{\text{pred}} - \phi_{\text{exp}})^2. \tag{7}$$

The number of iterations and parameter bounds for $\phi_{\text{v,sh}}$ and H_{corr} are provided in Table A.2.

The tunable parameters of the proposed framework are summarized in Table 1. Their selection and adjustment strongly depend on the available data, the underlying model, and the desired responsiveness of the CL behavior. Due to the large number of parameters, a structured tuning procedure is required.

The proposed approach is to first tune the novelty-related threshold parameters of the novelty filter. The MMD threshold is tuned to minimize false-positive activations when the PSD distribution has not significantly shifted and should therefore be evaluated using measurements of stationary distributions. The deviation threshold δ_{lim} is set to distinguish variability in experimental data from significant deviations in order to avoid unnecessary training steps. In the presented approach, δ_{lim} is derived from the maximum standard deviation of triplicate measurement errors of the solids mass fraction within the initial dataset. After tuning the novelty thresholds, the update frequency should be adjusted according to the process dynamics by selecting appropriate values for the novelty check time interval $\Delta t_{\text{novelty}}$ and the buffer size n_{buffer}. The minimal update interval is determined by multiplication of both parameters. If the novelty module is configured too conservatively, the responsiveness of the system is reduced. In contrast, overly aggressive parameter configurations may compromise model stability and introduce unnecessary fluctuations. Examples for different configurations of the novelty module are illustrated in Figs. A.1 and A.2. The selected parameters in Table 1 therefore represent a balance between responsiveness and stability. Furthermore, the training parameters, including the learning rate α and the number of epochs n_{epoch} must be tuned jointly to balance prediction accuracy and update latency. They should be selected as low as possible while maintaining stable convergence. The relatively high learning rate in this work is required to enable rapid adjustments with a low number of training epochs, thereby keeping computational cost minimal. This aspect is discussed in Section 5.2. Finally, Memory stability is fine-tuned using the parameters γ and λ_{EWC} once the prediction error has reached an acceptable level. Memory stability is evaluated using the prediction errors of previously seen data and unseen validation data, as presented in Sections 5.3 and 5.4.

All models and modules of the proposed CL workflow are implemented in Python.

4. Case study: Continual learning of Mill-Decanter process circuit data

The evaluation of the proposed workflow was conducted through an application-oriented case study based on a process circuit consisting of a laboratory-scale agitator ball mill and a laboratory-scale decanter centrifuge. This configuration allows a controlled shift of particle size while assessing the model's ability to adapt to changes in material properties. Mill-decanter-process circuits are applied in several industries, such as the ceramics and coatings, where finely dispersed particles with a narrow PSD are required. To achieve these PSDs, milling circuits are commonly used across multiple industrial sectors [30]. Such systems typically combine milling with classification, where the mill comminutes the particles and a classifier separates the material into coarse and fine fractions. Depending on the process configuration, the system operates in open- and closed-circuit configurations. In open-circuit operation, milling and classification occur sequentially without feedback. In contrast, closed-circuit operation increases yield by recycling the coarse fraction to the mill, while the fine fraction is withdrawn [31].

In the present work, only the decanter centrifuge is modeled, while the mill is treated as a black box. The PSD of its product stream is measured continuously using an online laser diffraction system and serves as an input to the GBM. During the process circuit experiments, both the feed PSD and feed solids mass fraction vary dynamically. Since the baseline GBM was trained exclusively on limestone suspensions with fixed material properties, prediction errors are expected when these material properties change. To mitigate this loss in predictive accuracy, the GBM is incrementally trained during the case study using the CL workflow described in Section 3.

4.1. Experimental setup

Fig. 3 shows the experimental setup used for the case study, including a laboratory-scale agitator ball mill (LabStar, NETZSCH-Feinmahltechnik GmbH, Germany) and a laboratory-scale decanter centrifuge (MD80, Lemitec GmbH, Germany) combined in a closed-circuit operation. The machine geometries are summarized in Table A.3. The stirred feed suspension is contained in the feed tank and is continuously pumped into a holding tank via a peristaltic pump to ensure a constant fill level in the holding tank. From the holding tank, the suspension is transferred by a second peristaltic pump into the grinding chamber of the mill, where it is subjected to comminution. Continuous measurement of the PSD is carried out using an online laser diffraction device (MYCELL, Sympatec GmbH, Germany) installed in the process line. For this purpose, a small partial stream of the milled suspension is diverted from the outlet of the mill to the laser diffraction device where it is autonomously diluted with water. Subsequently, the main suspension stream enters the decanter centrifuge, where particle

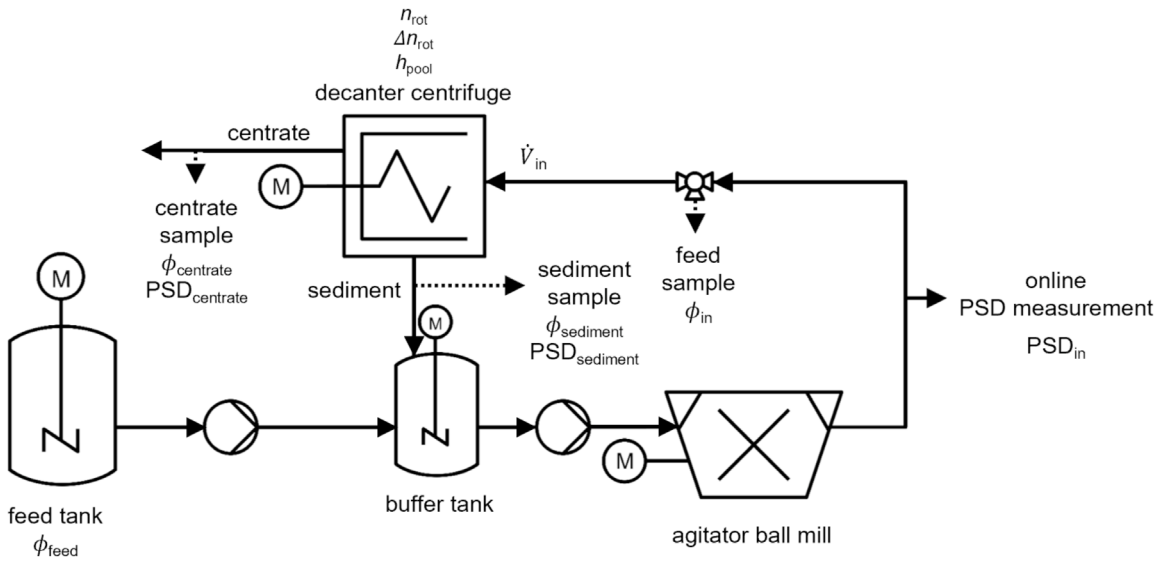

Fig. 3. Schematic setup of the process chain, including a decanter centrifuge, an agitator ball mill, and an online laser diffraction measurement instrument.

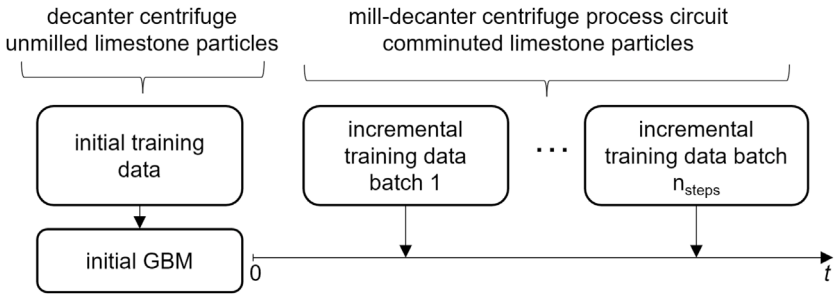

Fig. 4. Overview of the data used for model training. The initial GBM is trained on experimental data obtained from a laboratory-scale decanter centrifuge using unmilled limestone suspensions. Incremental online training is subsequently performed using data from mill-decanter process circuit experiments.

separation takes place. The solid-rich concentrate is returned to the stirred holding tank where it is mixed with the feed suspension from the feed tank in accordance with a closed grinding circuit. The centrate is withdrawn from the circuit.

Samples of the centrate and sediment streams are taken during the experiment for offline determination of the PSD and solids mass fraction by gravimetric measurement. In addition, the solids mass fraction of the feed stream is determined by sampling at a three-way valve positioned upstream of the decanter. During the experiments, the solids mass fraction of the feed stream to the decanter, the concentrate, and the centrate are recorded at intervals of approximately 3-4 min. Additionally, the PSD of the feed stream is captured every 10-20 s through online laser diffraction measurements.

The suspensions used in this study are composed of demineralized water as the continuous phase and limestone (LS) particles as the disperse phase with a density of 2700 kg m^{-3}. To stabilize the LS-water suspensions, 0.1 wt% sodium pyrophosphate was added to the suspensions. During the experiments, the suspensions were stirred continuously using an impeller stirrer to ensure homogeneous suspensions. Relevant process parameters for the experiments are summarized in Table 2.

4.2. Data structure

Fig. 4 illustrates the workflow of the presented case study. The initial GBM was trained on data obtained from experimental studies using unmilled LS-water-suspensions in the laboratory-scale decanter centrifuge (MD80, Lemitec GmbH, Germany). In total, 124 experimental datasets were used, covering combinations of process parameters including volumetric flow rate, feed solids mass fraction, rotational speed, and differential speed. For the process circuit data, the measured solids mass fraction in the feed, centrate and sediment streams as well as the temperature and PSD, which were recorded and measured at irregular intervals, are linearly interpolated. This is because the WBM requires discretized time-series inputs. Data normalization is performed using a MinMaxScaler for the features and a StandardScaler for the targets, both provided by the scikit-learn library.

5. Results and discussion

5.1. Mill-Decanter process circuits: Process dynamics and initial model performance

In this section, we evaluate the performance of the initial GBM as a baseline and discuss its limitations in capturing drifting material behavior, illustrated through mill-decanter process circuit experiments. Before evaluating model performance, the process behavior of the mill-decanter process circuit must be characterized to obtain a better understanding of the process dynamics. Fig. 5 shows the evolution of the solids mass fraction in the decanter feed stream for experiments A and B. Here, the process time t=0 min marks the time when the first sediment is discharged from the decanter centrifuge. Figs. 5(a) and (c) reveal an increasing solids mass fraction in the feed stream over time. This trend is caused by progressive solids build-up in the buffer tank due to continuous recirculation of the sediment stream.

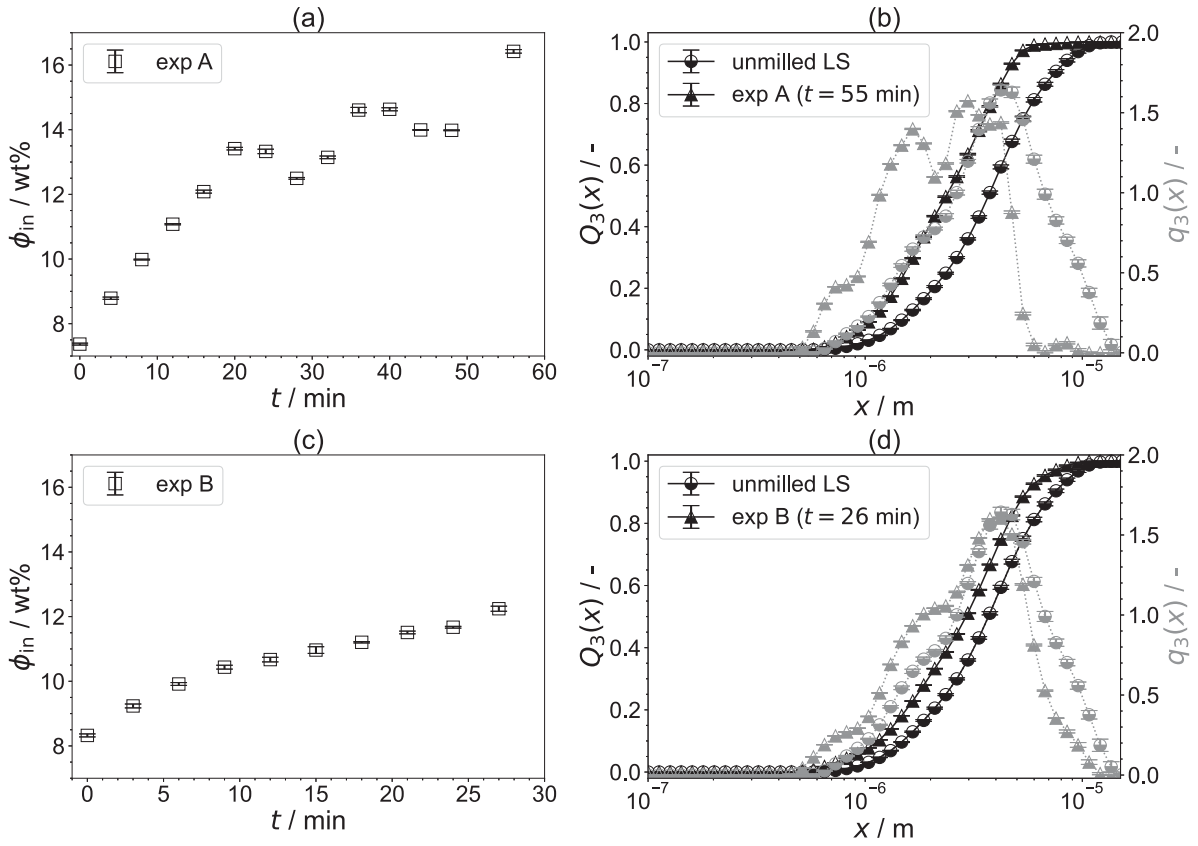

Fig. 5. Dynamic process behavior of the mill-decanter process circuit. Evolution of the solids mass fraction in the feed stream of the decanter centrifuge ϕ_{in} over the process time t for (a) experiment A and (c) experiment B. The corresponding cumulative volume-based distribution $Q_3(x)$ and density distribution $q_3(x)$ of the particle size x comparing unmilled limestone (LS) particles and comminuted LS particles are shown for (b) experiment A, and (d) experiment B.

After approximately t=20 min, experiment A exhibits fluctuations in the feed solids mass fraction, caused by the periodic sediment discharge behavior. Comparing the cumulative volume-based particle size distribution $Q_3(x)$ and the corresponding density distribution $q_3(x)$ in Figs. 5(b) and (d), a reduction in particle size is visible due to comminution in the mill. The influence of these changing feed properties on separation performance is reflected in the outlet streams in Fig. 6. In both experiments, the sediment solids mass fraction shows an initial strong increase that subsequently flattens. The centrate solids mass fraction in Figs. 6(b) and (d) initially increase in both experiments. This can be attributed to both the increasing feed solids mass fraction and the decreasing particle size, both of which hinder particle settling. In experiment A, fluctuations occur after t>24 min, corresponding to the fluctuations observed in the feed stream.

Evaluating the initial GBM without any further training as a baseline shows substantial deviations from the experimental results. As illustrated in Figs. 6(a) and (c), the model significantly overestimates the sediment solids mass fraction and produces predictions that remain nearly constant over time, only showing variations during the first 20 min. This highlights a clear mismatch between simulation and measurements. For the centrate solids mass fraction in Figs. 6(b) and (d), the initial GBM agrees reasonably well with the measured centrate solids mass fraction during the first 24 min. Beyond this point, however, increasing deviations appear, indicating a drift in settling behavior that the model cannot account for. In experiment B, the deviation is less pronounced, partly because the experiment duration is shorter.

Overall, the baseline results highlight a core limitation of the initial model. When material properties in the feed change during operation, the model performance significantly degrades. This motivates the need for continual learning approaches, enabling adaptations to shifting process conditions.

Table 2
Overview of the relevant process parameters for experiments A and B.

Process parameter		Exp A	Exp B	Dimension
Δn_{rot}	differential speed	25	25	min^{-1}
G	g-force	45	179	m s^{-2}
ϕ_{in} (t=0)	inlet solids mass fraction	7.37	8.33	wt%
$\dot{V}_{in}$	inlet volumetric flow rate	42	42	l h^{-1}
h_{pool}	pool depth	0.012	0.012	m
n_{rot}	rotational speed	1000	2000	min^{-1}
$v_{agitator}$	agitator tip speed	10	10	m s^{-1}
FD	filling degree	80	80	%

5.2. Continual learning performance

In order to assess the adaptability of the proposed framework, we perform incremental online training on the data from experiment A. Since the CL results are highly sensitive to regularization strength, models trained with different regularization strengths λ_{EWC} are evaluated and compared to a naive fine-tuned model and the initial model. The total number of incremental training steps varies between 23-27. Fig. 7 shows the trend of the predicted solids mass fraction during online adaptation.

Fig. 7(a) illustrates that the first incremental update significantly improves the sediment solids mass fraction prediction, correcting the initial overestimation through adjusting the compression function with the parameter $\phi_{v,sh}$. For high regularization strength, updates to previously optimized weights are strongly penalized, limiting model adaptation. Thus, the overestimation persists for λ_{EWC}=500 until late in the experiment. A similar trend is observed for the centrate solids mass fraction in Fig. 7(b). For t<30 min, the initial model agrees well

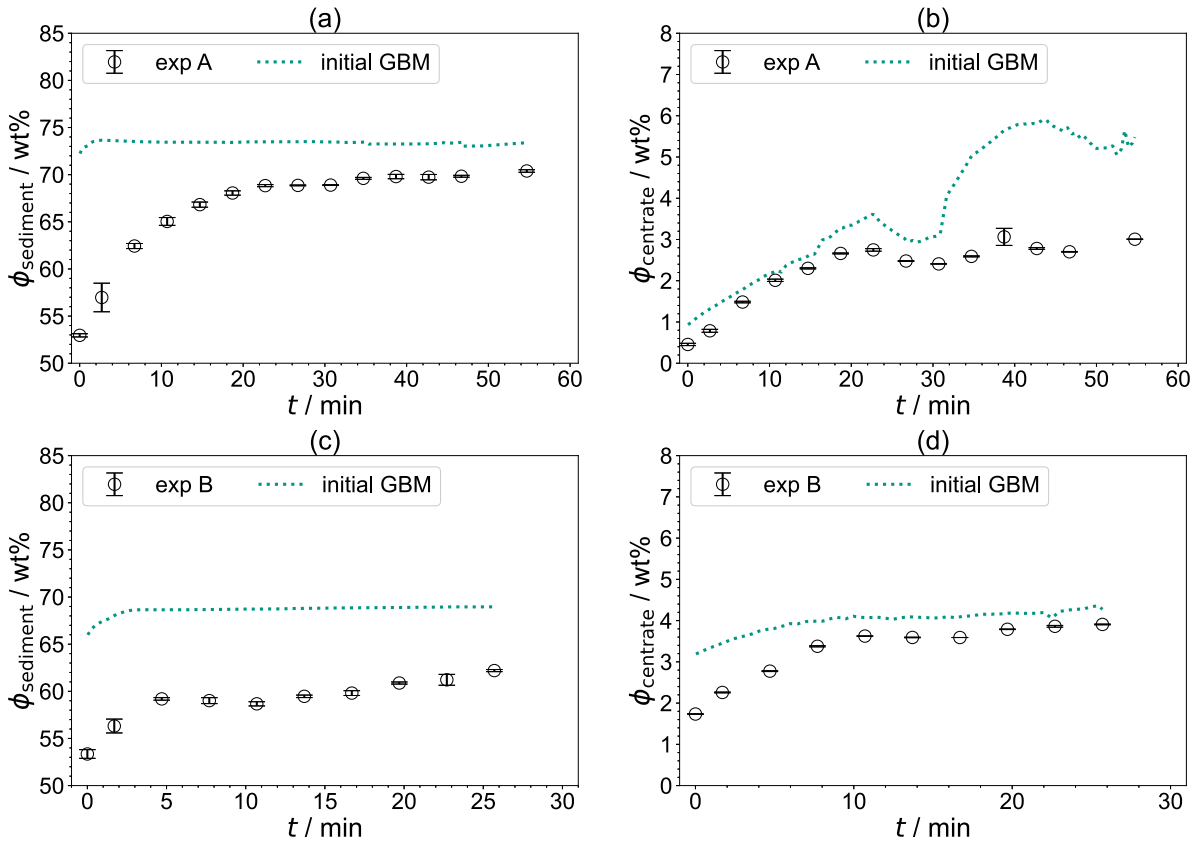

Fig. 6. Prediction of process circuit experiments using the initial GBM. The predicted sediment solids mass fraction $\phi_{sediment}$ and centrate solids mass fraction $\phi_{centrate}$ for experiment A are shown in (a) and (b), respectively, and for experiment B in (c) and (d), respectively. These predictions serve as a baseline for subsequent comparison and evaluation.

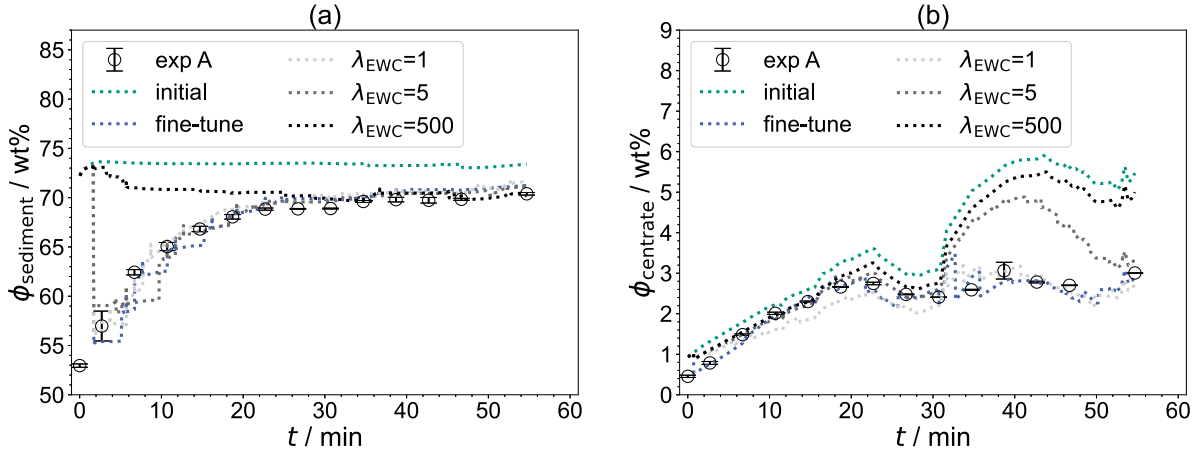

Fig. 7. Comparison of the online predictions of the solids mass fraction ϕ over time t for experiment A. Results for the initial model, naive fine-tuning, and EWC-regularized models with varying regularization strengths λ_{EWC} (see Eq. (3)) are shown for (a) sediment and (b) centrate. The epoch count is n_{epoch}=100.

with the experiments, and the updates have a minimal effect. Beyond t>30 min, the predictions start to drift, and only the fine-tuned model and the model with λ_{EWC}=1 are flexible enough to adapt effectively. Models with larger regularization strength fail to learn due to excessive penalization. This demonstrates the trade-off inherent in EWC. While higher regularization strength preserves memory, it limits the ability to learn new tasks. The prediction errors are summarized in Table 3. It confirms that raw adaptation is best for the naive fine-tuned model and moderate regularization strength (λ_{EWC}=1). With increasing regularization strength, the prediction error grows, indicating a lack of flexibility, although this measure does not account for memory stability and generalization ability, which will be discussed in Sections 5.3 and 5.4.

While the novelty module performs reliably on the available data, its robustness may be affected under high measurement noise. Static noise in the PSD is expected to largely cancel out when comparing

Table 3
RMSE for online prediction of experiment A. The first data point at t=0 is excluded from the RMSE calculation because no incremental training has been performed at this stage.

Model	$RMSE_{centrate}$/wt%	$RMSE_{sediment}$/wt%
initial	1.73	7.17
fine-tune	**0.16**	1.56
λ_{EWC} = 1	0.29	**0.95**
λ_{EWC} = 5	0.97	1.19
λ_{EWC} = 500	1.44	5.50

Table 4
Influence of the training epoch count n_{epoch} on the update latency $t_{latency}$, catch-up time $t_{catch\text{-}up}$, and RMSE with λ_{EWC}=1.

n_{epoch}/-	average $t_{latency}$/s	average $t_{catch\text{-}up}$/s	RMSE $\phi_{centrate}$/wt%	RMSE $\phi_{sediment}$/wt%
50	6.71 ± 0.42	0.78 ± 0.02	1.70	1.12
75	9.60 ± 0.40	1.20 ± 0.09	0.32	1.02
100	12.40 ± 0.44	0.92 ± 0.03	0.29	0.95
125	13.56 ± 0.88	1.55 ± 0.02	0.30	0.99
150	16.37 ± 0.52	1.53 ± 0.02	0.36	1.07
175	18.00 ± 1.23	2.59 ± 0.40	0.29	1.26
200	20.11 ± 1.78	2.84 ± 0.30	0.30	1.13

distributions measured with the same instrument, as noise characteristics remain consistent. However, issues arise when the noise magnitude approaches that of a true distribution shift or when MMD parameters are poorly tuned. Faulty input measurements may therefore trigger false-positive novelty detections, leading to unnecessary training steps that increases computational effort and can negatively impact memory stability. In contrast, noisy or faulty measurements of the output variable (solids mass fraction) directly affects the training signal and may degrade model performance. The use of online EWC mitigates this effect by constraining important weights and preventing large parameter updates from periodic faulty measurements. Nevertheless, frequent disturbances may still degrade model performance and memory stability. Future work should therefore investigate the impact of noisy and faulty measurements on the robustness of the novelty filter.

To assess the practical feasibility for online operation, we evaluated the number of epochs per incremental update on model latency and responsiveness. Fig. 8 shows the update latency, which describes the model delay due to an incremental learning step, as a function of the number of epochs. Lower update latency improves online feasibility, as it reduces the delay between data acquisition and model adaptation. As expected, increasing the number of epochs results in longer update times, rendering online application impractical beyond a certain threshold. Although predictive accuracy increased with each additional epoch, this improvement plateaued after 75-100 epochs. This indicates diminishing returns in accuracy relative to the additional computational cost and delay. Consequently, a moderate number of epochs provides the most suitable compromise between prediction accuracy, and online feasibility. In addition, Table 4 presents the average catch-up time, defined as the duration required by the model to resynchronize with real-time operation. This metric depends on both the update latency and the computational speed of the GBM. Although acceptable catch-up times are process-dependent, the obtained values are reasonably low, even for higher epoch counts, suggesting the proposed framework can maintain near-real-time operation under the tested conditions.[1]

It is important to note that the parameter identification via optimization contributes substantially to the update latency reported in Table 4. Using the settings given in Table A.2, the average optimization time per incremental update is t_{opt}=6.1 s ± 0.9 s. At low epoch numbers, this step dominates the total update latency $t_{latency}$, while its relative impact decreases as the number of epochs increases. Since this computational cost is specific to the presented modeling approach, it can be considered a static latency component. Reducing the number optimization iterations would decrease latency but at the expense of model accuracy.

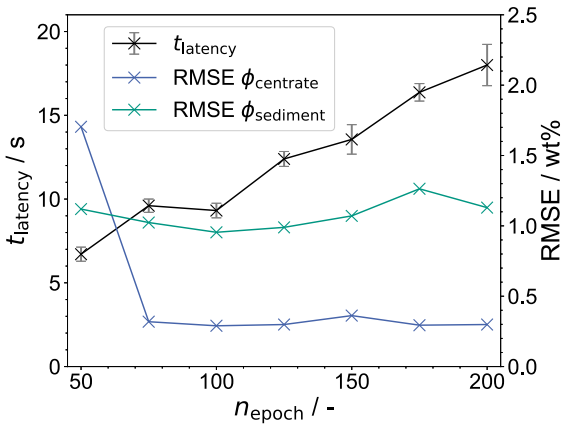

Fig. 8. Impact of the training epoch count n_{epoch} on the average update latency per training step $t_{latency}$ and on model prediction accuracy (RMSE) with λ_{EWC}=1.

5.3. Evaluation of catastrophic forgetting

In order to evaluate memory stability, the updated models are repeatedly evaluated on the same timeseries data as in Section 5.2 without any further training. Fig. 9 shows that the naive fine-tuned baseline diverges strongly from the experimental data for t<30 min, while predictions at later times remain accurate. This behavior indicates poor memory stability. In contrast, models with high regularization strength (λ_{EWC}>1) show limited adaptation and behave similarly to the initial GBM. Strong regularization thus preserves previous knowledge but restricts learning new tasks. Models with intermediate regularization (λ_{EWC}=1) achieve a better balance between plasticity and stability, preserving earlier knowledge better than the naive fine-tuned model while still adapting to new data. Similar trends are observed for the prediction of the centrate solids mass fraction shown in Fig. 9(b). While EWC does not completely eliminate forgetting, it significantly mitigates it compared to the fine-tuned model.

Fig. 10 summarizes the RMSE for experiment A, with the experimental timeline segmented into four time intervals. This segmentation allows assessment of catastrophic forgetting across the process, where early intervals correspond to previously learned tasks and later intervals reflect adaptation to newly encountered data. A model without forgetting should yield similarly low errors across all time intervals. The naive fine-tuned model mainly performs well only in later stages, confirming the occurrence of catastrophic forgetting. Models with moderate regularization strengths λ_{EWC}=1 and λ_{EWC}=5 exhibit more uniform errors, indicating improved memory stability. For high regularization strength, the high prediction error in earlier stages is caused by insufficient adaptation rather than forgetting, since the predictions remain close to the initial GBM.

Even though the naive fine-tuned model demonstrates strong short-term adaptability to new data in Section 5.2, its poor long-term memory poses a significant limitation for online operation. In real process environments, process conditions can exhibit cyclic patterns or temporary disturbances followed by a return to earlier states. A model that overwrites previous knowledge cannot reliably respond to such

[1] All simulations were run on the following hardware specifications: CPU: Intel Core i7-8700K @ 3.70 GHz, 6 cores; RAM: 64 GB DDR4; GPU: NVIDIA GTX 1080 Ti; OS: Microsoft Windows 11 using Python 3.11.

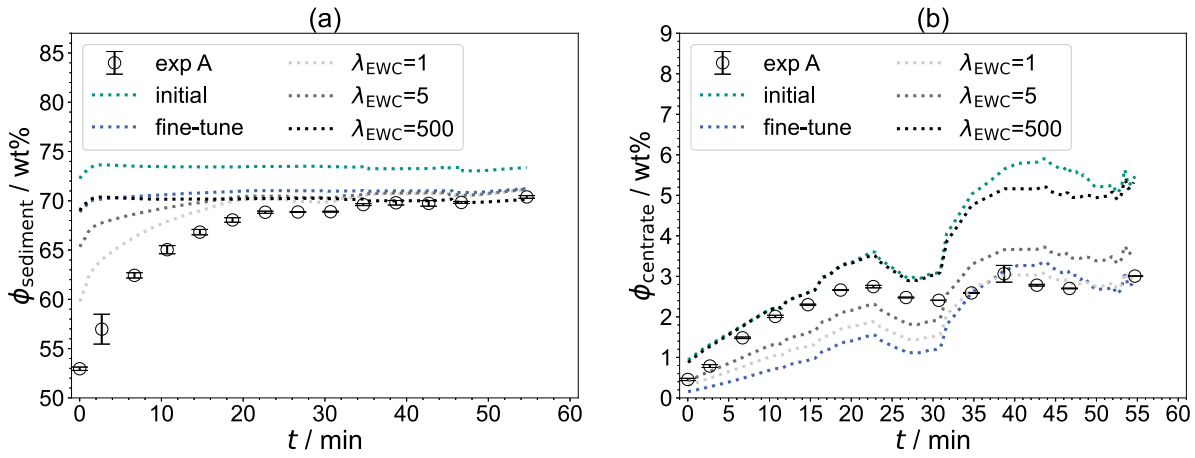

Fig. 9. Evaluation of memory stability for experiment A. Predictions of the solids mass fraction ϕ over time t are repeated after completing all incremental training steps to assess catastrophic forgetting. Results are shown for (a) sediment and (b) centrate.

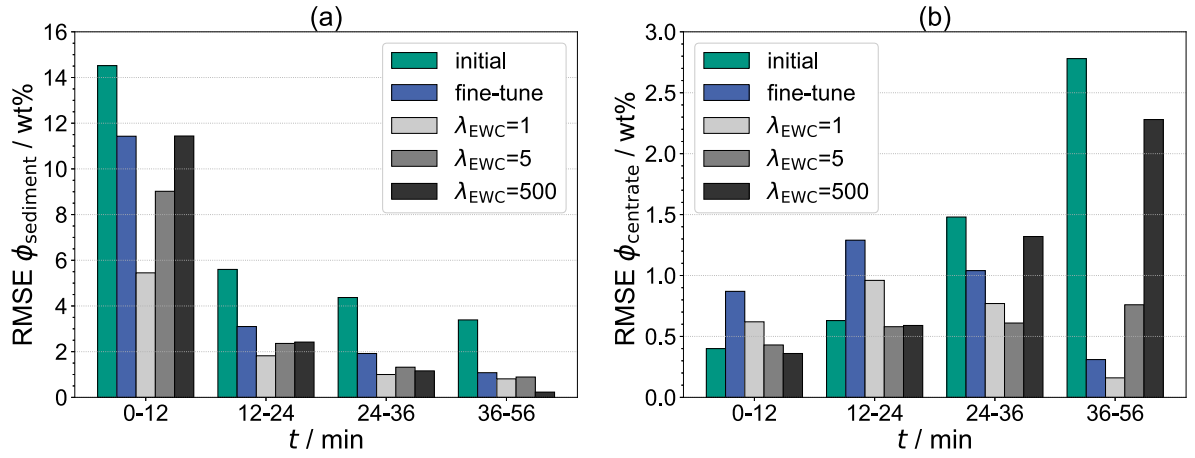

Fig. 10. RMSE for repeated prediction of experiment A. The experimental timeline is segmented into four time intervals to evaluate the occurrence and impact of catastrophic forgetting across different stages for the initial model, naive fine-tuning, and EWC-regularized models with varying regularization strengths λ_{EWC} (see Eq. (3)). Results are shown for (a) sediment solids mass fraction $\phi_{sediment}$ and (b) centrate solids mass fraction $\phi_{centrate}$.

conditions, leading to inconsistent predictions and reduced robustness. Maintaining a balance between adaptability and memory stability is thus critical for robust online CL.

5.4. Assessment of generalization with unseen experiment

For the evaluation of the generalization ability of the models, unseen data from experiment B is used without any further training. Fig. 11 compares the predictions of the naive fine-tuned baseline model, the initial GBM, and the EWC-regularized models with varying regularization strength.

Fig. 11(a) shows that the sediment solids mass fraction predicted by the fine-tuned model is clearly overestimated, yielding results comparable to those of the initial GBM. As discussed in Section 5.3, the naive fine-tuned model suffers from catastrophic forgetting. As a result, the model primarily retains the knowledge of the later stages (t>30 min) from experiment A, leading to poor generalization ability to the unseen operating conditions of experiment B. In contrast, the model with moderate regularization λ_{EWC}=1 exhibits close agreement with the validation data. However, performance is sensitive to the choice of regularization strength, as seen in Table 5. Excessive regularization causes the EWC-penalty term (see Eq. (2)) to dominate, resulting in model behavior similar to the initial GBM and limiting adaptability.

Fig. 11(b) shows the predictions for the centrate solids mass fraction and reveals a similar trend. The fine-tuned model consistently overestimates the centrate solids mass fraction over the process time, performing even worse than the initial GBM. This further illustrates the harmful impact of catastrophic forgetting on generalization performance. In contrast, the EWC-regularized models exhibit substantially improved agreement with the experiments.

Strong generalization capability is essential in industrial online learning scenarios, as it directly reduces the frequency and extent of required future model adaptations. Since catastrophic forgetting cannot yet be entirely avoided in CL systems, minimizing the need for model adaptation becomes critical for stable and computationally efficient operation. A model that generalizes well not only improves prediction accuracy on unseen process conditions but also enhances reliability against process variability, leading to a more reliable long-term deployment.

6. Conclusion

This paper investigates the performance and feasibility of a dynamic grey box model (GBM) for decanter centrifuges, integrated within an online Continual Learning (CL) framework for adaptive prediction under changing material properties and process behavior. By introducing

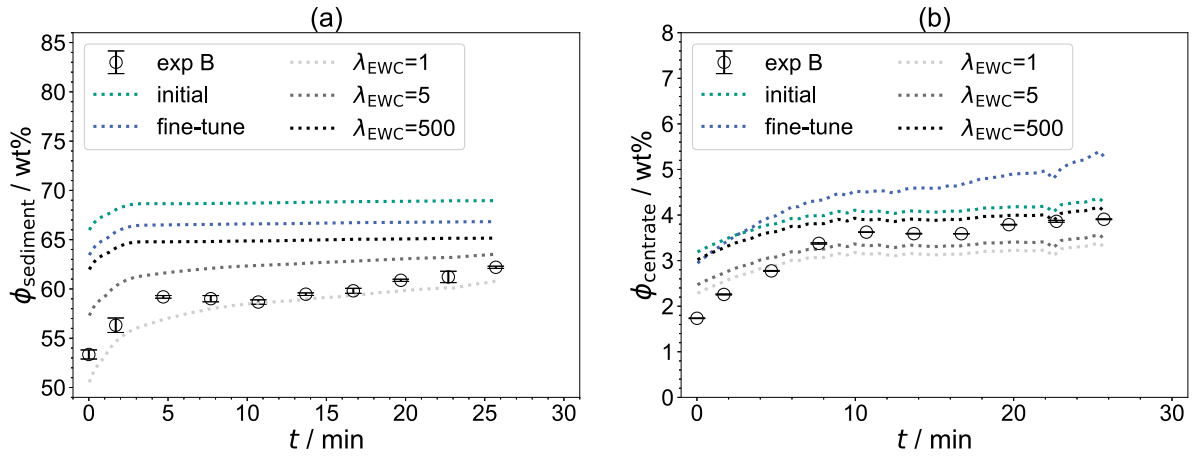

Fig. 11. Model validation on unseen experiment B. Predictions of the solids mass fraction ϕ over time t for the initial model, naive fine-tuning, and EWC-regularized models with varying regularization strengths λ_{EWC} are shown for (a) sediment and (b) centrate. No additional training was performed.

Table 5
RMSE for unseen experiment B.

Model	$RMSE_{centrate}$/wt%	$RMSE_{sediment}$/wt%
initial	0.86	9.92
fine-tune	1.09	7.42
$\lambda_{EWC} = 1$	0.48	**1.52**
$\lambda_{EWC} = 5$	**0.41**	2.98
$\lambda_{EWC} = 500$	0.63	5.81

CL to process engineering, this work addresses a research gap in which adaptive learning strategies remain underutilized despite increasingly dynamic industrial operating environments. The results demonstrate that EWC-regularized online adaptation improves prediction accuracy and mitigates performance degradation compared to naive fine-tuning, supporting effective model adaptation while preserving generalization. Consequently, prediction reliability is improved, enabling more efficient and stable decanter operation, optimized energy usage, and reduced operational costs.

Although the proposed approach shows strong potential for real-time industrial application, catastrophic forgetting is not entirely eliminated. Further research should therefore focus on robust hybrid CL strategies and tighter integration with reliable online measurement systems to improve long-term stability and resilience to noisy data. These advancements would strengthen the framework's suitability for model predictive control of decanter centrifuges. Beyond predictive control, the framework also holds potential as a decision-support tool for plant operators, enabling real-time recommendations and contributing to more adaptive process operation.

CRediT authorship contribution statement

Ouwen Zhai: Writing – original draft, Visualization, Validation, Methodology, Formal analysis, Data curation, Conceptualization. **Helene Rumiko Reppich:** Investigation. **Hermann Nirschl:** Writing – review & editing, Supervision, Resources, Funding acquisition. **Marco Gleiss:** Writing – review & editing, Supervision, Project administration.

Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Declaration of Generative AI and AI-assisted technologies in the writing process

During the preparation of this work, the authors used ChatGPT in order to improve the readability and language of the text. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the content of publication.

Acknowledgments

Open Access funding enabled and organized by Project DEAL. The authors acknowledge Lennard Ipe with the data collection.

Appendix A. Supplementary figures and tables

See Tables A.1–A.3 and Figs. A.1 and A.2.

Appendix B. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.seppur.2026.137951.

Table A.1
Architecture and hyperparameters of the MLP.

Parameter	Value	Dimension
activation function	sigmoid	–
number of epochs (initial training)	1000	–
number of neurons	60	–
L_2 regularization strength	10^{-6}	–
learning rate α (initial training)	10^{-2}	–
optimizer	adam	–

Table A.2
Parameter bounds and number of iterations of Bayesian optimization for parameter identification.

Parameter	Value
bounds $\phi_{v,sh}$	[−0.2, 0.2]
bounds H_{corr}	[0.6, 7]
number of iterations	8

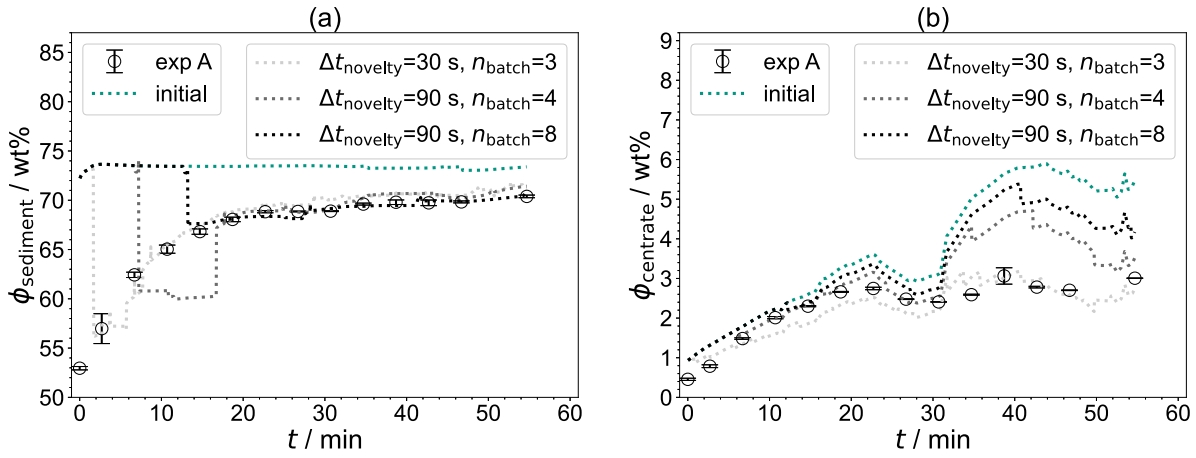

Fig. A.1. Comparison of the model responsiveness during online prediction for experiment A. Results for the initial model and EWC-regularized models (λ_{EWC}=1) with varying novelty check intervals $\Delta t_{novelty}$ and batch sizes n_{batch} are shown for (a) sediment and (b) centrate. Other framework parameters are set according to Table 1.

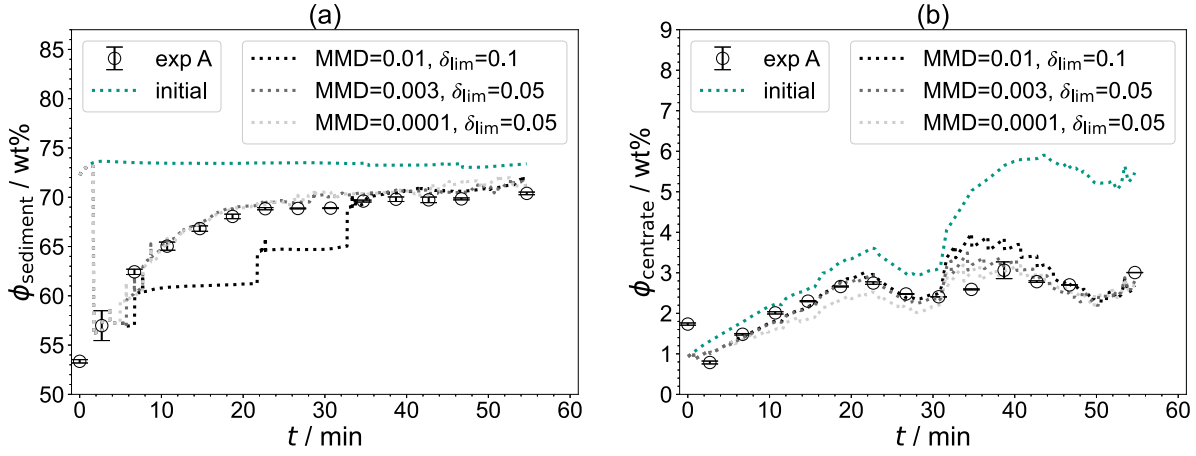

Fig. A.2. Comparison of novelty detection during online prediction for experiment A. Results for the initial model and EWC-regularized models (λ_{EWC}=1) with varying MMD and deviation thresholds δ_{lim} are shown for (a) sediment and (b) centrate. Other framework parameters are set according to Table 1.

Table A.3
Geometry of the laboratory-scale decanter centrifuge (MD80, Lemitec GmbH, Germany) and the laboratory-scale agitator ball mill (LabStar, NETZSCH-Feinmahltechnik GmbH, Germany).

MD80 geometry parameter	Value	Dimension
bowl radius	0.04	m
length cylinder	0.155	m
length cone	0.16	m
cone angle	7	°
LabStar geometry parameter	**Value**	**Dimension**
grinding chamber size	0.95	l
grinding balls diameter	2.0–2.2	mm

Data availability

Data will be made available on request.

References

[1] M. Zhu, D. Hu, Y. Xu, S. Zhao, Design and computational fluid dynamics analysis of a three-phase decanter centrifuge for oil-water-solid separation, Chem. Eng. Technol. 43 (5) (2020) 1005–1015, http://dx.doi.org/10.1002/ceat.201900245.

[2] H.K. Baust, S. Hammerich, H. König, H. Nirschl, M. Gleiß, Resolved simulation of the clarification and dewatering in decanter centrifuges, Processes 12 (1) (2024) 9, http://dx.doi.org/10.3390/pr12010009.

[3] C. Bai, H. Park, L. Wang, A model–based parametric study of centrifugal dewatering of mineral slurries, Minerals 12 (10) (2022) 1288, http://dx.doi.org/10.3390/min12101288.

[4] M. Gleiss, S. Hammerich, M. Kespe, H. Nirschl, Application of the dynamic flow sheet simulation concept to the solid–liquid separation: Separation of stabilized slurries in continuous centrifuges, Chem. Eng. Sci. 163 (2017) 167–178, http://dx.doi.org/10.1016/j.ces.2017.01.046.

[5] P. Menesklou, H. Nirschl, M. Gleiss, Dewatering of finely dispersed calcium carbonate-water slurries in decanter centrifuges: About modelling of a dynamic simulation tool, Sep. Purif. Technol. 251 (2020) 117287, http://dx.doi.org/10.1016/j.seppur.2020.117287.

[6] O. Zhai, N. Ehret, F. Rhein, M. Gleiss, Enhancing decanter centrifuge process design with data-driven material parameters in multi-compartment modeling, J. Adv. Manuf. Process. 6 (3) (2024) e10179, http://dx.doi.org/10.1002/amp2.10179.

[7] L. Wang, X. Zhang, H. Su, J. Zhu, A comprehensive survey of continual learning: Theory, method and application, IEEE Trans. Pattern Anal. Mach. Intell. 46 (8) (2024) 5362–5383, http://dx.doi.org/10.1109/TPAMI.2024.3367329.

[8] G.M. van de Ven, T. Tuytelaars, A.S. Tolias, Three types of incremental learning, Nat. Mach. Intell. 4 (12) (2022) 1185–1197, http://dx.doi.org/10.1038/s42256-022-00568-3.

[9] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A.A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, D. Hassabis, C. Clopath, D. Kumaran, R. Hadsell, Overcoming catastrophic forgetting in neural networks, PNAS 114 (13) (2017) 3521–3526, http://dx.doi.org/10.1073/pnas.1611835114.

[10] J. Schwarz, W. Czarnecki, J. Luketina, A. Grabska-Barwinska, Y.W. Teh, R. Pascanu, R. Hadsell, Progress & compress: A scalable framework for continual learning, in: Proceedings of the 35th International Conference on Machine Learning, PMLR, 2018, pp. 4528–4537.

[11] A. Robins, Catastrophic forgetting, rehearsal and pseudorehearsal, Connect. Sci. 7 (2) (1995) 123–146, http://dx.doi.org/10.1080/09540099550039318.

[12] D. Rolnick, A. Ahuja, J. Schwarz, T.P. Lillicrap, G. Wayne, Experience replay for continual learning, 2019, http://dx.doi.org/10.48550/arXiv.1811.11682, arXiv:1811.11682, [cs].
[13] J. Serrà, D. Surís, M. Miron, A. Karatzoglou, Overcoming catastrophic forgetting with hard attention to the task, 2018, http://dx.doi.org/10.48550/arXiv.1801.01423, arXiv:1801.01423, [cs].
[14] F. Zenke, B. Poole, S. Ganguli, Continual learning through synaptic intelligence, in: Proceedings of the 34th International Conference on Machine Learning, PMLR, 2017, pp. 3987–3995.
[15] Q. Yang, L. Wang, J. Wicker, G. Dobbie, Continual learning: A systematic literature review, Neural Netw. 195 (2026) 108226, http://dx.doi.org/10.1016/j.neunet.2025.108226.
[16] C. Baweja, B. Glocker, K. Kamnitsas, Towards continual learning in medical imaging, 2018, http://dx.doi.org/10.48550/arXiv.1811.02496, arXiv:1811.02496, [cs].
[17] M. Lenga, H. Schulz, A. Saalbach, Continual learning for domain adaptation in chest x-ray classification, 2020, http://dx.doi.org/10.48550/arXiv.2001.05922, arXiv:2001.05922, [cs].
[18] T. Lesort, V. Lomonaco, A. Stoian, D. Maltoni, D. Filliat, N. Dí az Rodríguez, Continual learning for robotics: definition, framework, learning strategies, opportunities and challenges, Inf. Fusion 58 (2020) 52–68, http://dx.doi.org/10.1016/j.inffus.2019.12.004.
[19] H. Tercan, A. Guajardo, T. Meisen, Industrial Transfer learning: Boosting machine learning in Production, in: 2019 IEEE 17th International Conference on INDIN, Vol. 1, 2019, pp. 274–279, http://dx.doi.org/10.1109/INDIN41052.2019.8972099.
[20] B. Maschler, H. Vietz, N. Jazdi, M. Weyrich, Continual Learning of Fault prediction for Turbofan engines Using Deep learning with Elastic weight consolidation, in: 2020 25th IEEE International Conference on ETFA, Vol. 1, 2020, pp. 959–966, http://dx.doi.org/10.1109/ETFA46521.2020.9211903.
[21] Y. Fu, A. Howard, C. Zeng, Y. Chen, P. Gao, P. Stinis, Physics-guided continual learning for predicting emerging aqueous organic redox flow battery material performance, ACS Energy Lett. 9 (6) (2024) 2767–2774, http://dx.doi.org/10.1021/acsenergylett.4c00493.
[22] Y. Balaji, M. Farajtabar, D. Yin, A. Mott, A. Li, The effectiveness of memory replay in large scale continual learning, 2020, http://dx.doi.org/10.48550/arXiv.2010.02418, arXiv:2010.02418, [cs].
[23] S.-A. Rebuffi, A. Kolesnikov, G. Sperl, C.H. Lampert, iCaRL: Incremental classifier and representation learning, 2017, http://dx.doi.org/10.48550/arXiv.1611.07725, arXiv:1611.07725, [cs].
[24] A.A. Rusu, N.C. Rabinowitz, G. Desjardins, H. Soyer, J. Kirkpatrick, K. Kavukcuoglu, R. Pascanu, R. Hadsell, Progressive neural networks, 2022, http://dx.doi.org/10.48550/arXiv.1606.04671, arXiv:1606.04671, [cs].
[25] R. Aoki, F. Tung, G.L. Oliveira, Heterogeneous multi-task learning with expert diversity, IEEE/ACM Trans. Comput. Biol. Bioinf. 19 (6) (2022) 3093–3102, http://dx.doi.org/10.1109/TCBB.2022.3175456.
[26] C. Fernando, D. Banarse, C. Blundell, Y. Zwols, D. Ha, A.A. Rusu, A. Pritzel, D. Wierstra, PathNet: Evolution channels gradient descent in super neural networks, 2017, http://dx.doi.org/10.48550/arXiv.1701.08734, arXiv:1701.08734, [cs].
[27] A. Chaudhry, P.K. Dokania, T. Ajanthan, P.H.S. Torr, Riemannian walk for incremental learning: Understanding forgetting and intransigence, in: V. Ferrari, M. Hebert, C. Sminchisescu, Y. Weiss (Eds.), in: Computer Vision – ECCV 2018, Vol. 11215, Springer, Cham, 2018, pp. 556–572, http://dx.doi.org/10.1007/978-3-030-01252-6_33.
[28] D.J.C. MacKay, A practical bayesian framework for backpropagation networks, Neural Comput. 4 (3) (1992) 448–472, http://dx.doi.org/10.1162/neco.1992.4.3.448.
[29] A. Gretton, K.M. Borgwardt, M.J. Rasch, B. Scholkopf, A. Smola, A kernel two-sample test, J. Mach. Learn. Res. 13 (25) (2012) 723–773.
[30] F. Müller, Wet classification in the fines range < 10 μm, Chem. Eng. Technol. 33 (9) (2010) 1419–1426, http://dx.doi.org/10.1002/ceat.200900540.
[31] P. Toneva, W. Peukert, Chapter 20 modelling of mills and milling circuits, in: Handbook of Powder Technology, Vol. 12, Elsevier Science B. V., Amsterdam, 2007, pp. 873–911, http://dx.doi.org/10.1016/S0167-3785(07)12023-6.