



Research
AI for Process Manufacturing—Article

Hedging Against Material Uncertainty via Chance-Constrained Recurrent Neural Networks: A Continuous Pharmaceutical Manufacturing Case Study



Qingbo Meng, I. David L. Bogle, Vassilis M. Charitopoulos*

Department of Chemical Engineering, Sargent Centre for Process Systems Engineering, University College London (UCL), Torrington Place, London WC1E 7JE, UK

ARTICLE INFO

Article history:

Received 16 August 2024

Revised 30 October 2024

Accepted 28 May 2025

Available online 18 August 2025

Keywords:

Data-driven chance constraints

Recurrent neural networks

Managing material uncertainty

Continuous pharmaceutical manufacturing

Smart manufacturing

ABSTRACT

In the pharmaceutical industry, model-based prediction is a crucial stage in process development that allows pharmaceutical companies to simulate different scenarios toward improving process efficiency, reducing costs, and enhancing product quality. Nevertheless, ensuring the quality of formulated pharmaceutical products through the management of raw material variations has always been a challenging task. In this work, data-driven chance-constrained recurrent neural networks (CCRNNs) are developed to address the issue arising from raw material uncertainty. Our goal is to explore how, by proactively incorporating uncertainty into the model training process, more accurate predictions and enhanced robustness can be realized. The proposed approach is tested on a fluid bed dryer (FBD) from a continuous pharmaceutical manufacturing pilot plant. The results demonstrate that CCRNN models offer more robust and accurate predictions for the critical quality attribute (CQA)—in this case, moisture content—when material variations occur, compared with conventional recurrent neural network-based models.

© 2025 THE AUTHORS. Published by Elsevier LTD on behalf of Chinese Academy of Engineering and Higher Education Press Limited Company. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

1. Introduction

As one of the most important industries in public health, with 1 trillion USD in annual sales [1], the pharmaceutical industry is currently experiencing a drastic transformation toward more intelligent and responsive processes in order to withstand rapidly changing conditions, including unpredictable market demand, raw material variations, environmental conditions, and the accelerated emergence of novel technologies. There is widespread recognition of the importance of developing and implementing more efficient and intelligent manufacturing processes to address unforeseen changes and ensure high product quality and profitability [2]. The primary challenge is to make manufacturing processes more robust, flexible, and responsive in order to improve the ability to effectively address these uncertain changes and reduce their impact on product quality [3–6]. A variety of studies have implemented different methodologies, such as machine learning and artificial intelligence, to develop surrogate models that can achieve critical quality attribute (CQA) prediction or con-

trol while maintaining product quality. However, these studies have not explicitly considered the presence of uncertainty in the model training process. The objective of the present work is to explore how, by proactively incorporating uncertainty into the model training process, more accurate predictions with increased robustness and a reduction in uncertainty propagation can be realized.

Uncertainty is ubiquitous across the chemical and pharmaceutical industries. As Sharifian et al. [7] have pointed out, uncertainties are typically attributed to a deficiency in knowledge regarding a specific topic or situation, which is particularly true for the pharmaceutical industry due to complex chemical and physical reactions across multiple material phases within batch and continuous reactor units. The potential impacts caused by a lack of consideration of uncertainty, such as inconsistent product quality, safety risks, and inaccurate testing results, have been reported by the U.S. Food and Drug Administration (FDA; USA) [8]. Moreover, the decision-making process for design and operation is significantly affected by the presence of uncertainties, which can lead to suboptimal outcomes and increased unexpected risks.

For a continuous pharmaceutical manufacturing process, surrogate-based techniques are effective methods to reduce product variations when uncertainty and experimental noise are

* Corresponding author.

E-mail address: v.charitopoulos@ucl.ac.uk (V.M. Charitopoulos).

considered [9]. More recently, mathematical programming techniques have been introduced to address the challenges arising from uncertainty issues, offering positive solutions through the application of methodologies such as data-driven modeling techniques [10], stochastic programming [11–14], robust optimization [15–18], chance-constrained optimization [19,20], and their data-driven variants. In this study, data-driven chance-constrained programming is selected to tackle material uncertainty due to one particular advantage over robust optimization and stochastic programming: it allows constraints to be satisfied with a user-defined risk level or probability. This feature offers a flexible approach to uncertainty management and facilitates a trade-off between process reliability and operability.

Additionally, to achieve accurate model-based CQA prediction, artificial neural networks (ANNs) have received increasing attention and been successfully applied in the pharmaceutical industry as a cutting-edge methodology because of their ability to capture nonlinear relationships between inputs and outputs without knowledge of the underlying mathematical relationships [21,22]. The multilayer perceptron (MLP), as a standard feedforward ANN model, has been widely adopted for the effective mapping of complex input–output relationships [23,24], key variable predictions [25–28], and process optimization tasks where statistical model development is required [29–32]. It has also been applied to develop surrogate models to investigate the prediction variance obtained via an adaptive sampling statistical approach and to improve the accuracy of the identified feasible region [33]. However, due to their “black box” nature, MLPs do not offer causal explanations for events that occur within the system [34]. Furthermore, the absence of a feedback loop in MLP networks’ structure limits their ability to model sequential or time-series data effectively. Instead, recurrent neural networks (RNNs) have been proposed by incorporating at least one feedback loop to enable bi-directional signal propagation [35]; this allows the model to learn from previous experience, exhibiting memory-like behavior and enhancing its ability to solve dynamic problems [36]. RNNs are usually employed to replace a dynamic process plant model and thereby reduce the computational effort associated with the tasks of process dynamic prediction or control [37–39]. Alternatively, they are employed in the prediction of various pharmacokinetic parameters, such as drug absorption, bioavailability, or distribution volume [40]. Convolutional neural networks (CNNs) are mostly used in image recognition and computer vision [41] because of the way they engage convolutional layers for feature extraction [42]. CNNs have found fewer applications than the other two in the pharmaceutical industry, where most focus is on image-

based objectives, such as drug target interaction [43]. However, although there are plenty of successful applications of ANNs in the pharmaceutical industry, the presence of uncertainty in the model training processes remains largely unexplored. The contribution of this work is to investigate how, by proactively accounting for uncertainty during the training process of RNNs, the stability and robustness of their prediction performance can be enhanced.

In this work, the intra-particle void fraction (IVF) of lactose, which can cause material quality variations in pharmaceutical particulate products, is considered the uncertain variable. We propose a methodology that utilizes data-driven chance-constrained programming to quantify uncertainty across various risk levels, integrating these constraints during the training process of RNNs. Compared with traditional RNNs, chance-constrained RNNs (CCRNNs) provide a framework for incorporating uncertainty into time-series predictions and decision-making processes, making them particularly valuable in applications where risk management and robustness are critical. This approach enhances the standard RNN’s capabilities by allowing it to operate effectively under material uncertainty, ultimately leading to improved prediction performance and reliability.

The case study used here to test our method is the fluid bed dryer (FBD) that is part of the tableting manufacturing process (Fig. 1) at the Diamond Pilot Plant (DiPP) at the University of Sheffield, from which the data were obtained. Ultimately, the prediction error and violation rate of the reliability targets are recorded as the assessment criteria for evaluating the prediction performance of the alternative RNN models.

The remainder of this paper is structured as follows: Section 2 details the dynamic process model and sets out the key computational elements of the proposed approach. In Section 3, we test the proposed approach on a case study from the continuous pharmaceutical manufacturing domain and present the results. Lastly, Section 4 provides concluding remarks and outlines ongoing research directions.

2. Process model and methodology

2.1. Dynamic model description

The segmented FBD (red dashed rectangle in Fig. 1) is a commonly employed unit in pharmaceutical industrial settings for removing the moisture from granular materials. It has several notable advantages, including reduced drying time, a high heat transfer coefficient [44,45], the generation of a uniform dryness distribution [46], and a low pressure drop. Moreover, its suitability

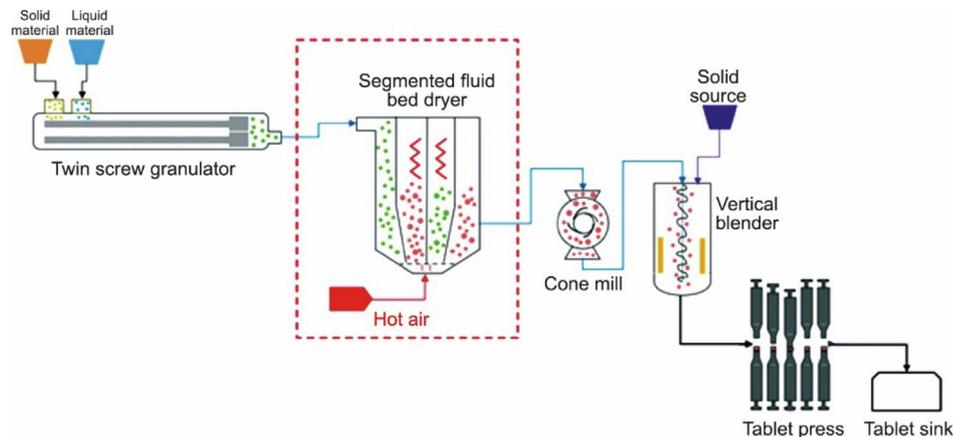


Fig. 1. Flowsheet of the continuous pharmaceutical tableting process.

extends to continuous systems, making it a versatile choice for various applications [47]. The FBD has the following processing steps:

(1) Granule feeding: Well-mixed wet granules are fed into the FBD chamber in sequence.

(2) Hot air distribution: Hot air is blown into the chamber from the bottom at a controlled velocity and is passed through a distribution plate, which helps to evenly distribute the hot air across the bed of granules.

(3) Fluidization: Hot air flows through the granules, which causes them to behave like a fluid as they are lifted and suspended in the air stream. The granules' moisture is effectively removed and taken away via moisture-laden air from the dryer through a filter system.

(4) Temperature and air flowrate control: During the drying process, the inlet air temperature and flowrate are carefully controlled to ensure the granules are dried uniformly without overheating or becoming damaged. The FBD is designed with a cooling zone, where cooler air is allowed to pass through the granules to bring down their temperature before discharge.

(5) Discharge: Once the drying process is completed, the granules are discharged from the FBD for further processing, such as milling, blending, or tablet compression.

The FBD model used in this study is adapted from a study by Burgschweiger and Tsotsas [48], to which interested readers are directed for more details on the solid and vapor mass and energy balance equations for the FBD model. A twin-screw granulator (TSG) is used to mix the feeding material (lactose) and liquid binder (water) to produce granules for the FBD drying process (Fig. 1). Several output variables of the TSG process are considered as input data for training the FBD model.

The population balance model for predicting the granule attributes is described below [49]:

$$\frac{\partial}{\partial t} n(\mathbf{v}, \mathbf{t}) + \frac{\partial}{\partial \mathbf{t}} \left[n(\mathbf{v}, \mathbf{t}) \frac{d\mathbf{v}}{d\mathbf{t}} \right] = B(\mathbf{v}, \mathbf{t}) - D(\mathbf{v}, \mathbf{t}) + F_{in} - F_{out} \quad (1)$$

where $n(\mathbf{v}, \mathbf{t})$ represents the granule number density function; \mathbf{v} is the granule volume vector; \mathbf{t} is time; and the functions $B(\mathbf{v}, \mathbf{t})$ and $D(\mathbf{v}, \mathbf{t})$ are utilized to denote the occurrence of particle birth and death resulting from breakage and nucleation processes, respectively. F_{in} and F_{out} correspond to the inlet/outlet flowrates in the TSG, respectively. The platform used for the digital model implementation is gPROMS FormulatedProducts (v. 2022.1.0, Siemens, Germany). The parameters were validated using DiPP at the University of Sheffield, and more details can be found in Ref. [50]. There are four manipulated inputs in total: the lactose and water flowrate feeding to the TSG, and the vapor flowrate and temperature entering the FBD. The granule moisture content (CQA) that exits the FBD is the predicted output.

2.2. Data-driven chance constraints

Compared with conventional chance-constrained programming, data-driven chance-constrained programming provides more flexibility when the underlying probability distribution of the uncertain parameters is not explicitly given. The distribution information can be estimated from historical process data, rather than purely relying a predetermined probability distribution or being based on an expert's experience. An integral component in applying data-driven chance constraints is the selection of a confidence level, which is designed to encompass the unknown or uncertain true distribution within a specified tolerance for the product quality variable [20,51]. Typically, a chance-constrained program involves a set of probabilistic constraints, as shown in Eq. (2):

$$P\{g_j(x) \geq \delta_j\} \geq 1 - \alpha_j, \quad j = 1, 2, \dots, m \quad (2)$$

where P represents the probability, and δ_j and $g_j(x)$ are the uncertain input variables and functions (constraints) of the decision variable x , respectively. The term $(1 - \alpha_j)$ represents the confidence sets, and α_j is also known as the risk level of the j th constraint, and $m \in \mathbf{N}^+$. Eq. (2) expresses that the probability of the j th uncertain variable satisfying the j th constraint must be at least $(1 - \alpha_j)$. A kernel density estimation approach is used to find the inverse cumulative density function (also known as the quantile function) of the uncertain variable used to determine a $g_j(x)$ that can satisfy the chosen level of probability for all possible occurrences of the uncertain variable δ_j . The mathematical expression of the reformulation is given by Eq. (3):

$$g_j(x) \geq F_{\delta_j}^{-1}(1 - \alpha_j), \quad j = 1, 2, \dots, m \quad (3)$$

where $F_{\delta_j}^{-1}$ is the quantile function. For more details, we refer interested readers to the works of Jiang and Guan [52] and Li et al. [53] for in-depth expositions of the topic.

2.3. RNN—long short-term memory (LSTM)

Traditional RNNs are well-suited for sequence-based or time-series data process modeling because the internal feedback loop architecture enables the model to learn from previous experience to give it memory-like behavior. The folded structure, which provides a concise view of information flows through the network, and the unfolded structure, which shows the explicit connections within the folded structure, are shown in Fig. 2, where u_t and y_t represent inputs and outputs at the t th time stamp, respectively. w_u , w_y , and w_f are the weights associated with the connection, and f_a stands for the activation function. It should be noted that the output y_t not only takes u_t as input, but also considers u_{t-1} as input:

$$y_t = w_y \cdot f_a(u_t \cdot w_u + w_f \cdot f_a(w_u \cdot u_{t-1})) \quad (4)$$

RNNs can be used to develop predictive models from data, but a common pitfall in their training performance in practice [54] is the vanishing and exploding gradient problem experienced during the backpropagation training process [55]. The gradient vanishing and exploding problem refers to the challenge in which gradients either diminish significantly or grow exponentially as they are backpropagated through long-term time steps because of the repeated multiplication of weighting matrices in the neural network architecture, leading to gradient values that either shrink to zero or explode to extremely large values. If the gradient values are too small, the training process becomes less effective and time-consuming; when they are too large, the training becomes unstable, causing difficulties in converging to the global minimum. This issue can be effectively addressed by utilizing LSTM, a special-

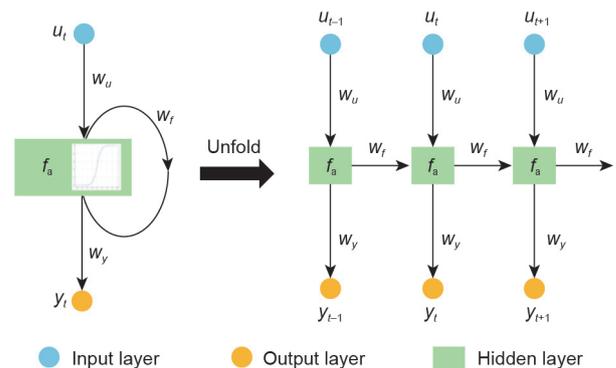


Fig. 2. The folded and unfolded structure of an RNN.

ized RNN architecture designed to retain information over long time periods [56], mitigating short-term memory limitations during the training process [57].

In the LSTM structure, the cell state c (Fig. 3) allows information to flow through the entire layer while controlling the flow of information added or removed at each time step [58]. This helps the LSTM unit retain long-term dependencies and remember relevant information over extended sequences, making it particularly effective for tasks involving sequential data processing. Fig. 3 illustrates the flow of data at time t . The notations used in Fig. 3 are summarized as follows:

- f_t , g_t , i_t , and o_t are the forget gate, cell candidate, input gate, and output gate, respectively;
- U_f , U_g , U_i , and U_o are input weights;
- R_f , R_g , R_i , and R_o are recurrent weights;
- b_f , b_g , b_i , and b_o are bias;
- σ and τ represent sigmoid and tanh activation functions, respectively.

The output h_t , also known as the hidden state, is computed using the following four steps:

Step 1: decide what information is to be removed from the cell state in the forget gate f_t :

$$f_t = \sigma(x_t U_f + R_f h_{t-1} + b_f) \quad (5)$$

where x_t means inputs at time t .

Step 2: determine what new information is to be stored in the cell state:

$$g_t = \sigma(x_t U_g + R_g h_{t-1} + b_g) \quad (6)$$

$$i_t = \tau(x_t U_i + R_i h_{t-1} + b_i) \quad (7)$$

Step 3: update the cell state c_t :

$$c_t = f_t c_{t-1} + g_t i_t \quad (8)$$

Step 4: compute the final output h_t :

$$o_t = \sigma(x_t U_o + R_o h_{t-1} + b_o) \quad (9)$$

$$h_t = o_t \tau(c_t) \quad (10)$$

LSTM blocks are linked together to construct the LSTM layer architecture, as shown in Fig. 4, where the input is $X = [x_1, x_2, \dots, x_t, \dots, x_T]$, the corresponding output is $Y = [y_1, y_2, \dots, y_t, \dots, y_T]$, and t denotes the time steps. It should be

noted that the output Y is equivalent to the hidden state h . This LSTM layer can obtain both the full sequence output Y and the final time step (T) output y_T .

2.4. Bi-directional LSTM

Bi-directional LSTM (Bi-LSTM) is an improved version of the LSTM architecture that incorporates both forward and backward directions for processing sequential data. During the training, inputs are fed into forward and backward LSTM blocks simultaneously to compute the hidden state and update the memory cell. This architecture (Fig. 5) enables the layer to capture past and future information at the same time in a single time step, thereby enhancing its ability to capture complex dependencies in the input sequence, which is not possible with the standard LSTM formulation [59].

2.5. Overfitting

Although the LSTM/Bi-LSTM structure effectively avoids vanishing and exploding gradient issues, it remains susceptible to overfitting during training. To deal with overfitting, we incorporate “dropout” layers into the neural network architecture. “Dropout” is a widely used stochastic regularization technique in the neural network training process [60]. At each training iteration, nodes (neurons) are stochastically deactivated by multiplying their outgoing signals by 0, with a desired probability rate to temporarily disable the nodes along with their connections [61]. This approach successfully prevents a situation in which the final outputs strongly rely on certain nodes [60,61]. Fig. 6 shows an example of a general neural network structure graph before and after the black node is dropped.

2.6. Proposed RNN architecture and workflow chart

Fig. 7 describes how the layers link together to form an RNN architecture. The standard LSTM and Bi-LSTM layers are employed together to solve long-term memory problems, as the training performance shows a smaller validation root mean square error (RMSE) value than when each layer is implemented individually (Table 1). There are two fully connected layers engaged in the proposed structure. The first one, in the middle, is used to extract essential features and patterns from the high-dimensional output of the Bi-LSTM layer. The second one, inserted before the output

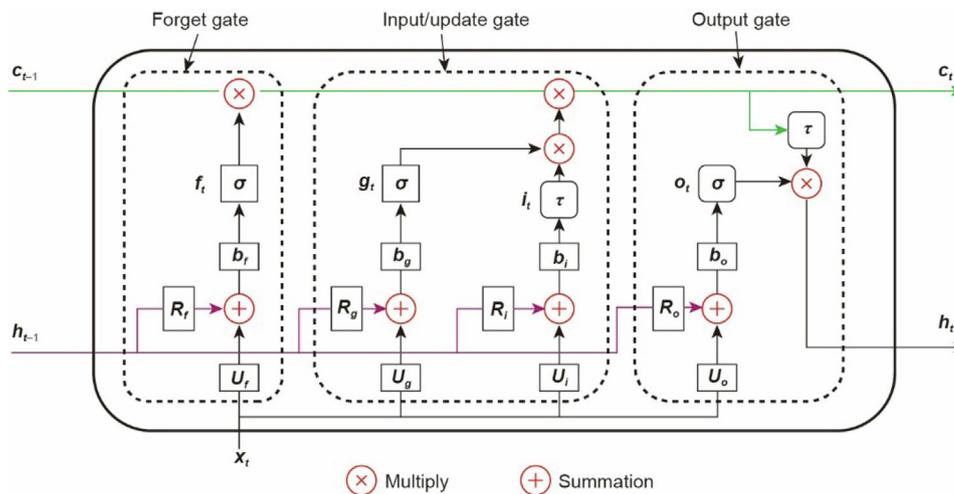


Fig. 3. An LSTM block.

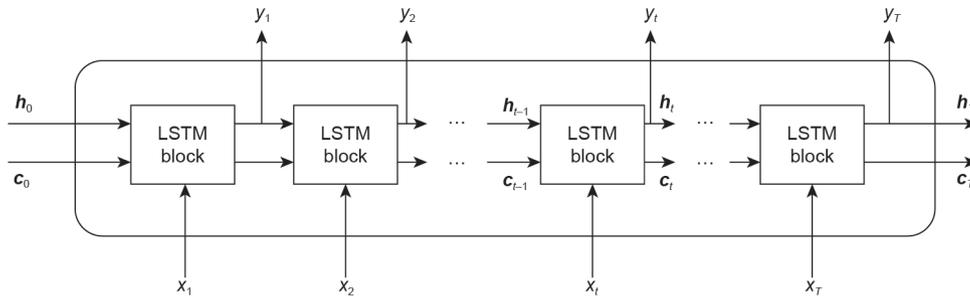


Fig. 4. LSTM layer.

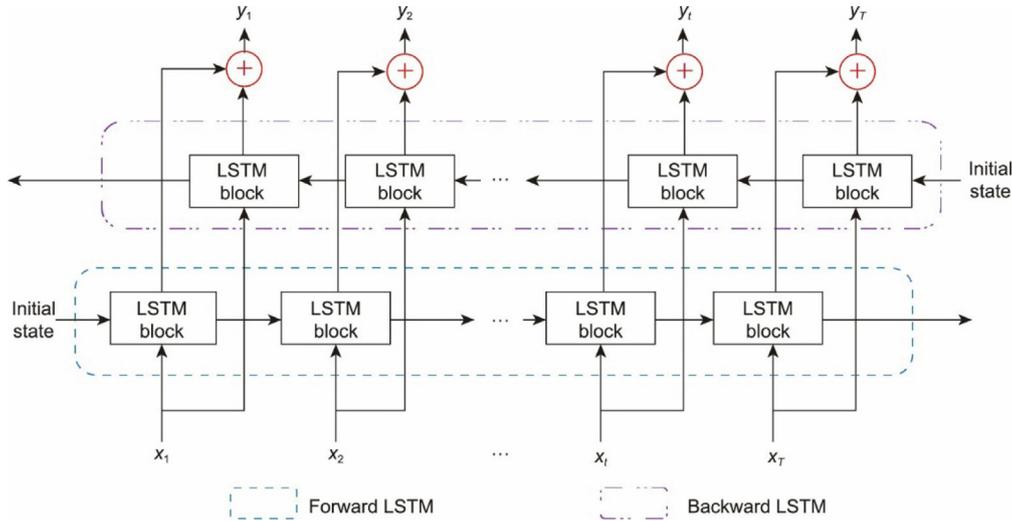


Fig. 5. Bi-LSTM structure.

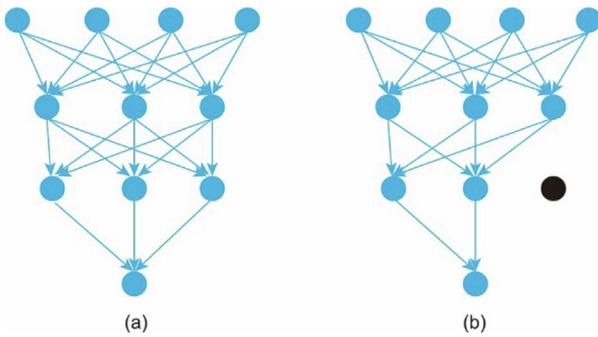


Fig. 6. Neural network structure (a) before and (b) after applying dropout.

layer, performs the prediction task that allows the layer to map the learned features to the appropriate output values.

The workflow chart is indicated in Fig. 8. Steps 2 and 3 are repeated to obtain datasets for the RNN model training in gPROMS FormulatedProducts. Further details are discussed in Section 3.1. Datasets are then exported to MATLAB (R2022b, MathWorks, USA), where the RNN model training, validating, and testing take place.

2.7. Prediction accuracy evaluation criteria

In this study, the RMSE is used to evaluate the RNN models' prediction accuracy performance. RMSE measures the prediction pre-

cision by squaring the errors between the predicted value and the observed data value to provide a positive score within $[0, \infty]$. A lower RMSE value indicates a closer alignment between the prediction outcome and the observed data. RMSE is defined as follows:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - y_i^{\text{pred}})^2}, i = 1, 2, \dots, n \quad (11)$$

where y_i and y_i^{pred} represent the observed data and predicted value of the i th sample, respectively; and n is the sample size. To provide more direct and meaningful results, the normalized RMSE (NRMSE) is introduced, based on the standard RMSE, and is given below:

$$\text{Normalized RMSE} = \frac{RMSE \text{ value}}{y_i^{\text{max}} - y_i^{\text{min}}}, i = 1, 2, \dots, n \quad (12)$$

where y_i^{max} and y_i^{min} are the maximum and minimum observed output values at the i th sample, respectively, and $y_i^{\text{max}} - y_i^{\text{min}}$ represents the output value range. NRMSE provides a relative measure of the error compared with the range of the data. This can be useful in understanding the proportion of error relative to the variability in the dataset.

Another criterion for evaluating prediction performance is the violation rate, as defined in Eq. (13).

$$\text{Violation rate: } R_{\text{vio}} = \frac{\text{count}(\text{Predicting RMSE value} \geq \text{Accepted criteria value})}{\text{Total sampled testing datasets}} \quad (13)$$

It is expressed as a ratio to quantify when the predicted RMSE value exceeds the expected criterion. A lower rate value implies better prediction performance.

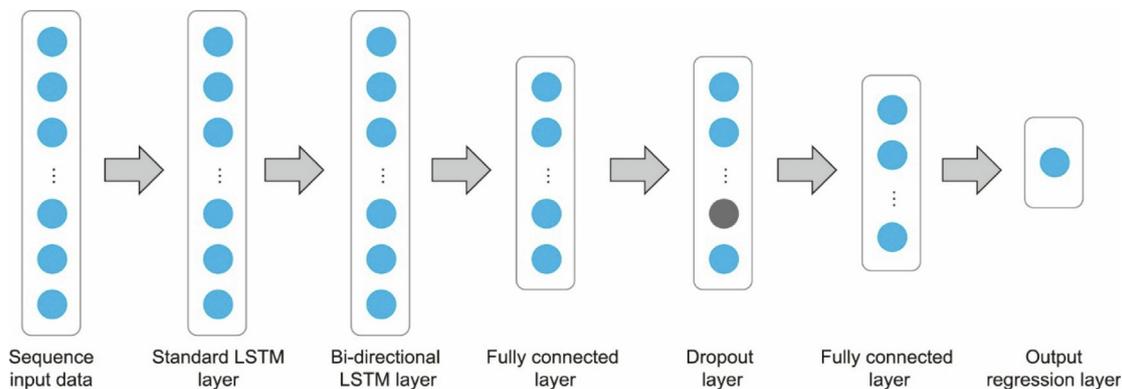


Fig. 7. RNN layers in the proposed architecture.

Table 1 Training performance with different combinations of LSTM and Bi-LSTM layers.

Engaged layer	Validation RMSE	Training time
LSTM only	0.088	165 min 52 s
Bi-LSTM only	0.083	182 min 29 s
LSTM + Bi-LSTM	0.071	235 min 4 s

Table 2 Input variable and boundary details.

Input variable	Process unit	Lower bound	Upper bound	Unit
Lactose feed flowrate	TSG	10	20	kg·h ⁻¹
Binder feed flowrate	TSG	2	6	kg·h ⁻¹
Vapor feed flowrate	FBD	280	380	m ³ ·h ⁻¹
Vapor temperature	FBD	40	80	°C

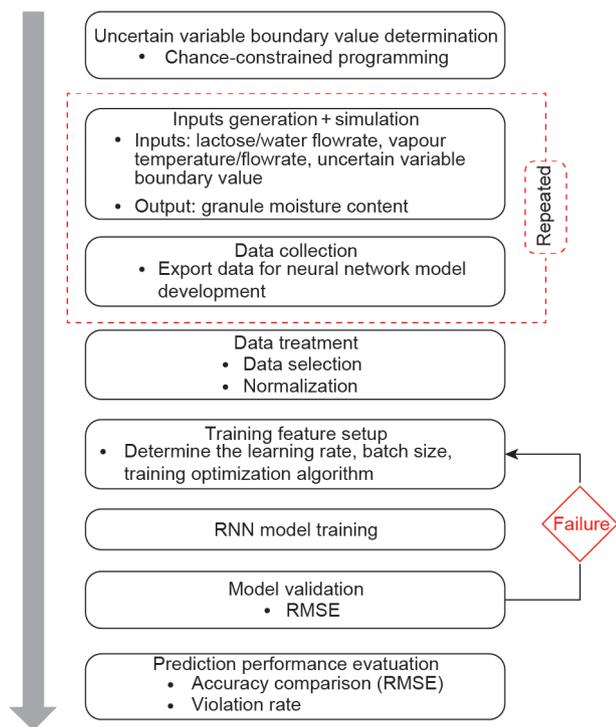


Fig. 8. Proposed workflow chart.

3. Case study

3.1. Simulation specification and data collection

The TSG and FBD model were developed in gPROMS FormulatedProducts. Input details for running the gPROMS model are described in Table 2. The boundary conditions for the simulation inputs were selected according to two perspectives:

(1) A previously published study by Ref. [50] on the TSG and FBD of the DiPP, which is utilized in this study, focuses on determining the key parameters through a model-driven design

approach, among other studies. For example, the powder feed rate (10–25 kg·h⁻¹) had a significant effect on the granule size in the TSG, so the boundary for this variable is between 10 and 20 kg·h⁻¹. A higher liquid/solid (L/S) ratio leads to an increased d_{50} and a decreased d_{10} , while the granule size distribution becomes narrower and more monomodal. When selecting the L/S ratio, it is better to include broader distributed information data for RNN model development; therefore, the boundary is chosen to be within the range 0.1–0.6, which results in the binder feed flowrate being maintained between 2 and 6 kg·h⁻¹.

(2) The second perspective involves the DiPP operation limits, such as the vapor temperature. The four segmented chambers in the FBD are sequentially fed with wet granules from the TSG for the setup loading time (240 s), drying time (700 s), and discharging time (50 s) of each segment. The whole drying process simulation time is 1650 s. High-temperature vapor is generated from the vapor source and pumped into the bottom of the FBD. Wet granules in segments are suspended in a hot air stream, and moisture is evaporated to the desired target during the drying process.

To capture the dynamic behavior of the system across different temporal states, the total feeding time is divided into various time intervals, with different input values for each interval. This allows the model to learn how the system evolves over time and how the inputs influence its behavior at various stages (Table 3).

In this study, we consider the lactose IVF as the only uncertain material variable. As there is only one uncertain variable in this problem, index j is equal to 1 in Eqs. (2) and (3). The IVF value data is collectable from the material property document and is assumed to follow a normal distribution with a mean of 0.5 and a standard deviation of 0.05 ($N \sim (0.5, 0.05)$). The kernel density estimation technique is then applied to obtain the quantile function for determining the boundary values of the uncertain variable IVF at each risk level—in other words, the probability that a randomly selected IVF value within the specified tolerance must meet the desired confidence levels of 90%, 95%, and 99%, respectively. The steps for how the boundary risk level is determined are given below:

(1) Generate sample data of uncertain variable IVF values that follow the specified normal distribution $N \sim (0.5, 0.05)$ in Fig. 9(a).

(2) Apply kernel density estimation techniques to estimate the probability density function (PDF) of the generated data.

Table 3
Time intervals of input variables.

Input variable	Total feeding time (s)	Number of time intervals	Duration of each time interval (s)
Lactose feed flowrate	960	16	60
Binder feed flowrate	960	16	60
Vapor feed flowrate	1600	8	200
Vapor temperature	1600	8	200

(3) Integrate the PDF to obtain the quantile function (Fig. 9(b)), which is used to calculate the corresponding IVF values at a user-selected risk level.

The mathematical formulation details of kernel density estimation have been described in our previous work [62] and are not repeated here for the sake of brevity.

Fig. 9(a) displays the histogram information of the uncertain variable IVF, while Fig. 9(b) illustrates the quantile function, along with the corresponding IVF values of 0.426, 0.407, and 0.355, which are the values for the risk levels $\alpha = 0.1, 0.05,$ and $0.01,$ respectively. These IVF values are set in the gPROMS digital model, initially for dataset collection to train and test RNN models of the FBD. It is important to note that the uncertain variable IVF does not need to conform strictly to a normal distribution; it can follow any type of distribution. By adhering to the computational steps outlined above, the corresponding IVF values at each risk level can be computed, regardless of any specific distribution of the uncertain variable.

Three data-driven CCRNN models, one for each risk level, and one standard RNN model are respectively developed in Table 4 (NA: not available). The model-based simulation datasets for the training and testing of the RNN models are generated by the following steps:

(1) Generate 100 groups of input variables within their operational lower and upper bounds in each time interval.

(2) Set up the three boundary IVF values (0.426, $\alpha_j = 0.10;$ 0.407, $\alpha_j = 0.05;$ and 0.355, $\alpha_j = 0.01$) determined according to Eq. (3) and one default value (IVF = 0.1 as a standard RNN model for the purpose of comparison).

(3) Run the gPROMS simulation to get 100 training datasets for each IVF value.

Following the above steps, we can obtain four groups of datasets, each containing 100 data samples. For each data sample, there are 19 dimensions of the inputs fed to the FBD, including the vapor inlet temperature and flowrate, lactose and granule mass flowrate, initial granule moisture content, porosity, and 13 particle size dis-

tributions, and one output—namely, the granule moisture content exiting the FBD. In each group, 80% and 20% of the datasets are randomly chosen for RNN model training and testing, respectively. Further details are provided in Figs. 9 and 10. MATLAB is the digital platform for the FBD RNN models' training and testing.

3.2. RNN model training and testing

The RNN input sequence layer and output layer contain 19 feature neurons and one responding neuron, respectively, in accordance with the input and output dimensions. The fully connected layer in the middle of the structure has 10 neurons, which is enough to learn the complex patterns from the input data while reducing the overfitting risk.

Due to the lack of a universally accepted theory for selecting the exact number of hidden units and training epochs, we decided to try several combinations of hidden units and epochs to observe the performance (Fig. 11). The results demonstrated that using 80 hidden units and 200 training epochs effectively minimizes the RMSE during the training process.

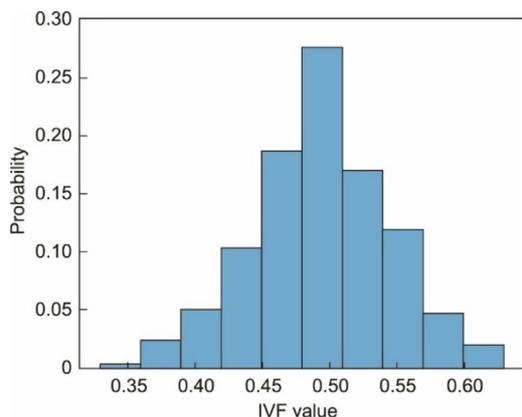
Table 1 shows the training performance when different combinations of LSTM and Bi-LSTM layers are applied in the structure. We note that the validation RMSE is reduced when both the LSTM and Bi-LSTM layers are employed in the neural network architecture; however, this requires slightly more computational time compared with using a single LSTM or Bi-LSTM layer. This is expected because adding more layers to the RNN structure increases the complexity of the model. Other key training features are listed in Table 5.

Three CCRNN models and one RNN model were developed after training and were named based on their probability confidence levels (Table 4). The moisture content upon exiting the FBD was identified as a CQA to be predicted using the four RNN models.

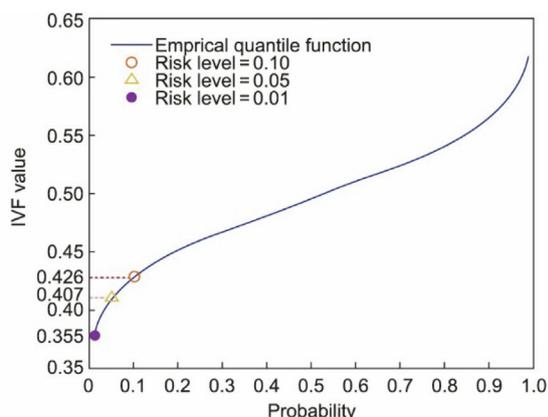
NRMSEs were computed to evaluate each RNN model's prediction accuracy and robustness in the presence of material quality variations. Three tests using different datasets were conducted for each respective RNN model's moisture content prediction per-

Table 4
RNN model variants with corresponding IVF values.

Model name	IVF value	Confidence level/risk level
NOCC (deterministic)	0.1 (default)	NA
P90	0.426	90%/10%
P95	0.407	95%/5%
P99	0.355	99%/1%



(a)



(b)

Fig. 9. (a) Histogram of IVF values follows a distribution $N \sim (0.5, 0.05)$; (b) boundary IVF values at different risk levels.

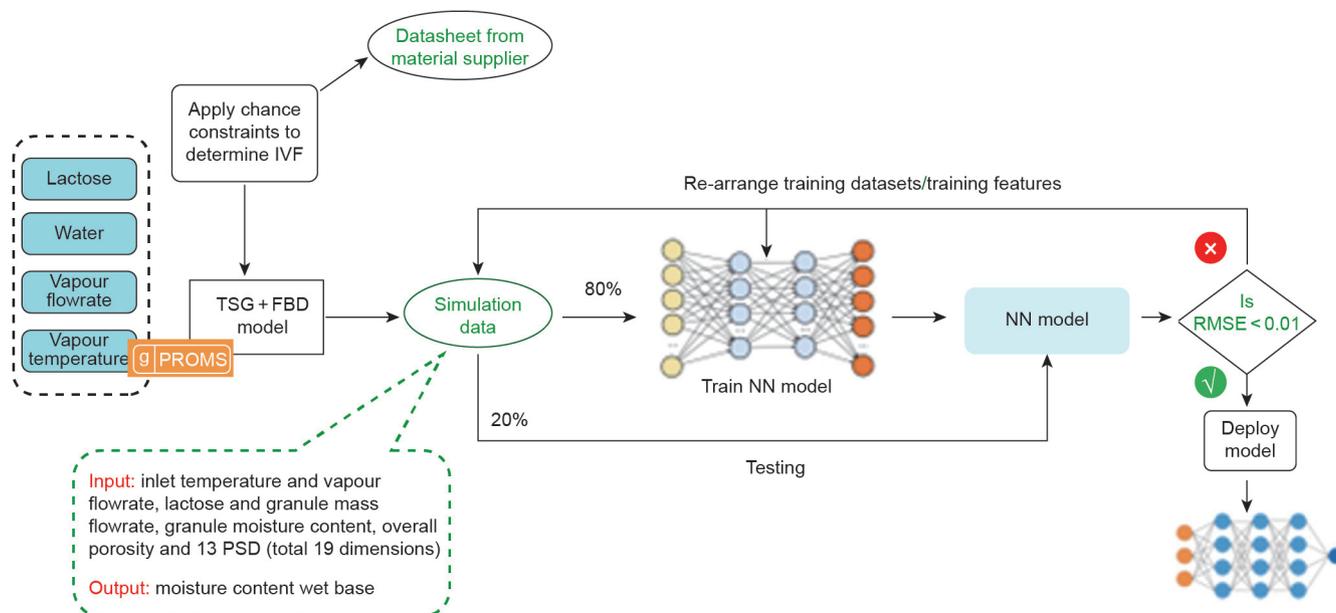


Fig. 10. RNN model development diagram. NN: neural network.

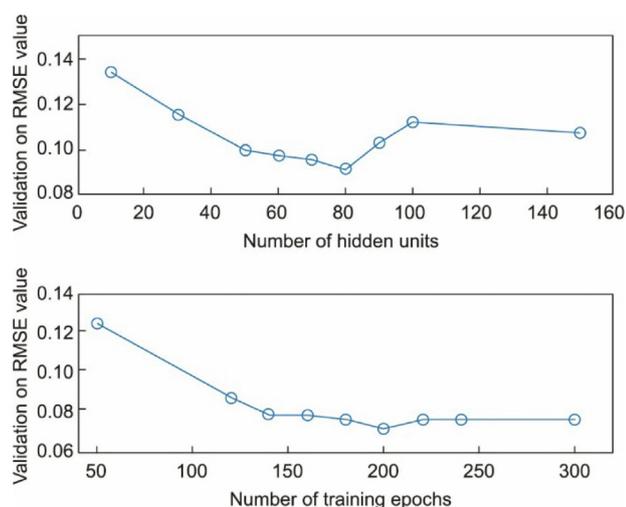


Fig. 11. Training validation RMSE values at different combinations of hidden units and training epochs.

Table 5
Training features for the RNN models.

Training features	Value/option
Mini batch size	1
Initial learning rate	0.01
Learn rate schedule	Piecewise
Solver name	Root mean square propagation (RMSProp)
Learn rate drop period	20
Learn rate drop factor	0.8
Regularization	L2 (0.02)
Validation frequency	25

formance evaluations. The purpose of running Test 1 was to investigate the RNN model's prediction performance on the unseen data. Tests 2 and 3 were designed to mimic the manufacturing plant scenarios in which the uncertain variable—the IVF of the material—was either unknown/random (Test 2) or conformed to a specific distribution (Test 3), based on its typical range between 0.3 and

0.5 in chemical manufacturing processes. Test 4 was designed to investigate whether alternative distributed (right-skewed) IVF values with the same mean and standard deviation as in Test 3 could impact the prediction performance of the CCRNNs. The testing data are described below:

- **Test 1:** Use the 20% testing datasets (Fig. 10) generated from each IVF value as inputs to the four well-trained RNN models in sequence to observe each model's prediction performance (Fig. 12).
- **Test 2:** Produce 50 alternative datasets from gPROMS with 50 randomly selected IVF values between [0.1, 0.6].
- **Test 3:** Produce 50 alternative datasets from gPROMS with 50 distributed IVF values that follow the normal distribution $N\sim(0.35, 0.1)$ between [0.1, 0.6].
- **Test 4:** Produce 50 alternative datasets from gPROMS with 50 distributed IVF values that follow the right-skewed (log-normal) distribution $G\sim(0.35, 0.1)$ between [0.1, 0.6].

4. Results

Boxplots are utilized to demonstrate the Test 1 results, as illustrated in Fig. 13. Each boxplot shows the distribution of the moisture content predicted by each RNN model visually, to make comparison easier. To provide a more direct and meaningful prediction accuracy, all RMSE values have been normalized and are displayed as a percentage. In this case study, an acceptable value of the RMSE percentage is when it is less than 4%. Fig. 13(a) shows that the deterministic (NOCC) model can achieve accurate predictions, indicated by all NRMSE values being less than 4% at IVF = 0.1. This is expected, as IVF = 0.1 was utilized to generate datasets for training the NOCC model. However, its predictive capability significantly deteriorates when applied to other groups of datasets. It was observed that two outliers were nearly double the desired acceptable criterion, measuring around 7%–8% when the IVF values were 0.426, 0.408, and 0.355. Meanwhile, the median values were very close to the desired value of 4%. These results indicate that the NOCC model demonstrates diminished predictive accuracy in the presence of material uncertainty.

However, Fig. 13(d) illustrates that the CCRNN model P99 has the best prediction performance for all group datasets, as it ensures

that the NRMSE values are consistently below 4%. The other two CCRNN models, P90 and P95 (Figs. 13(b) and (c)), also exhibit excellent predictive behavior across all testing datasets, being characterized by significantly lower NRMSE percentages compared with NOCC. Notably, when IVF = 0.100, only one NRMSE (Table 6) slightly exceeds the 4% criteria in a total of 20 testing datasets for P90 (5.32%) and P95 (4.4%).

The results of Test 1 imply that the CCRNN models demonstrate greater resilience and adaptability when confronted with unfamiliar datasets. Consequently, they are capable of mitigating material uncertainties to maintain the prediction performance at the intended level. Furthermore, when compared with the NOCC model, the CCRNNs provide relatively more accurate prediction performance, with a lower mean NRMSE and a narrower distributed range. Further NRMSE information is given in Table 6.

In Tests 2, 3, and 4, the violation rate, defined in Eq. (13), is employed to evaluate the prediction performance of the RNN models. It is noted that an acceptable NRMSE percentage should ideally be below 4%, while the predicted moisture content value range ($y_i^{\max} - y_i^{\min}$) estimated via the datasets is 0.25. Based on Eq. (12), the RMSE value is calculated to be 0.01. In other words, if we can maintain the predicted RMSE value at less than 0.01, the NRMSE

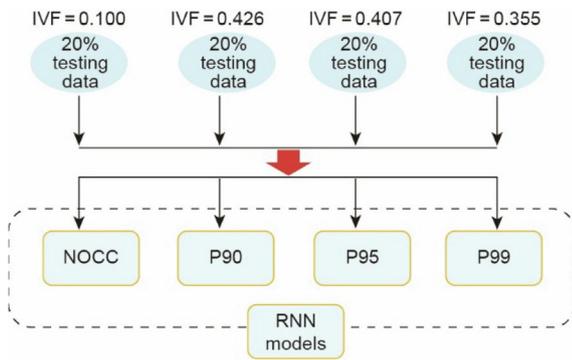


Fig. 12. Conceptual flowchart of Test 1.

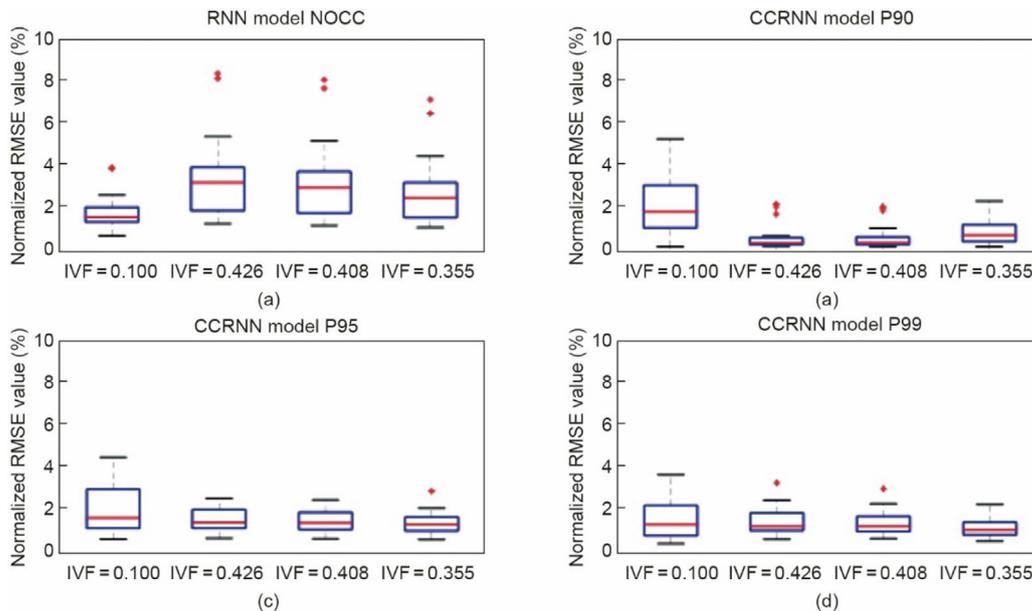


Fig. 13. Moisture content prediction RMSE boxplots of four RNN models under each IVF value.

can be guaranteed to be below 4%. Therefore, for all three tests, predicted RMSE values greater than 0.01 are counted as violations.

Figs. 14 and 15 show the RMSE histogram information of Tests 2 and 3, respectively. Both figures reveal that NOCC demonstrates the worst prediction performance, with the highest incidence of violations (a predicted RMSE ≥ 0.01) among the four RNN models. The quantitative results (Table 7) show that the violation rates predicted by the NOCC model are 32% and 34%, respectively. This reveals the same result as Test 1, which is that the NOCC model is unable to predict accurately if there are material variations. The other three CCRNN models show significant reductions in violation rate compared with NOCC. P99 performs the best among the three CCRNN models, with violation rates of only 4% and 2%, followed by P95, which has a violation rate of 6% in both Test 2 and Test 3. P90 is the worst among the three CCRNN models. More quantitative details are given in Table 7.

Fig. 16 illustrates the prediction performance of datasets generated using 50 IVF values that follow a log-normal distribution (Test 4). This test is designed to investigate whether the alternative input IVF distribution could impact the CCRNNs' prediction performance. From a theoretical perspective, input data that closely resembles the training data is more likely to produce accurate prediction results. This implies that, in this context, a greater number of IVF values that fall within the boundary range established by the chance-constrained programming based on the uncertain variable distribution information are likely to yield higher predictive accuracy. Consequently, the format of the input IVF distribution may not be the primary factor influencing moisture content predictions. This conclusion is supported by the results of Test 4, which show no significant changes in the violation rate when comparing the proposed right-skewed (log-normal) distributed IVF values with those from the normal distributed (Test 3) and the randomly selected IVF values (Test 2). The quantified violation rate data of Test 4 are presented in Table 7.

An investigation into the influence of various probability confidence intervals on the prediction RMSE is shown in Figs. 17 and 18. It can be seen that, as the probability risk level decreases, the predicted RMSE distribution curve becomes narrower and shifts to the left of the x-axis. This observed shift and shape transformation signifies a substantial enhancement in the prediction performance

Table 6
NRMSE values of RNN models' prediction.

Model	IVF	Min	Max	Mean	Model	IVF	Min	Max	Mean
NOCC	0.100	0.72%	3.92%	1.80%	P95	0.100	0.68%	4.40%	2.04%
	0.426	1.28%	8.32%	3.44%		0.426	0.72%	2.52%	1.56%
	0.407	1.20%	8.04%	3.24%		0.407	0.68%	2.44%	1.48%
P90	0.355	1.12%	7.08%	2.80%	P99	0.355	0.64%	2.84%	1.44%
	0.100	0.32%	5.32%	2.20%		0.100	0.44%	3.64%	1.60%
	0.426	0.36%	2.28%	0.76%		0.426	0.64%	3.24%	1.48%
	0.407	0.32%	2.16%	0.80%		0.407	0.68%	3.00%	1.36%
	0.355	0.32%	2.44%	1.04%		0.355	0.56%	2.24%	1.16%

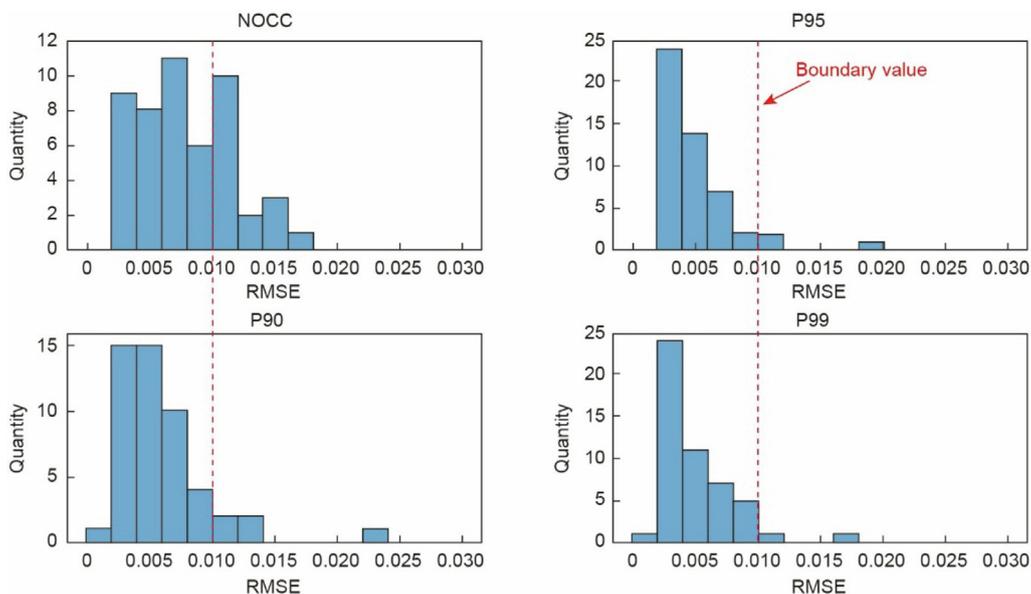


Fig. 14. Prediction performances of datasets generated using 50 randomly selected IVF values.

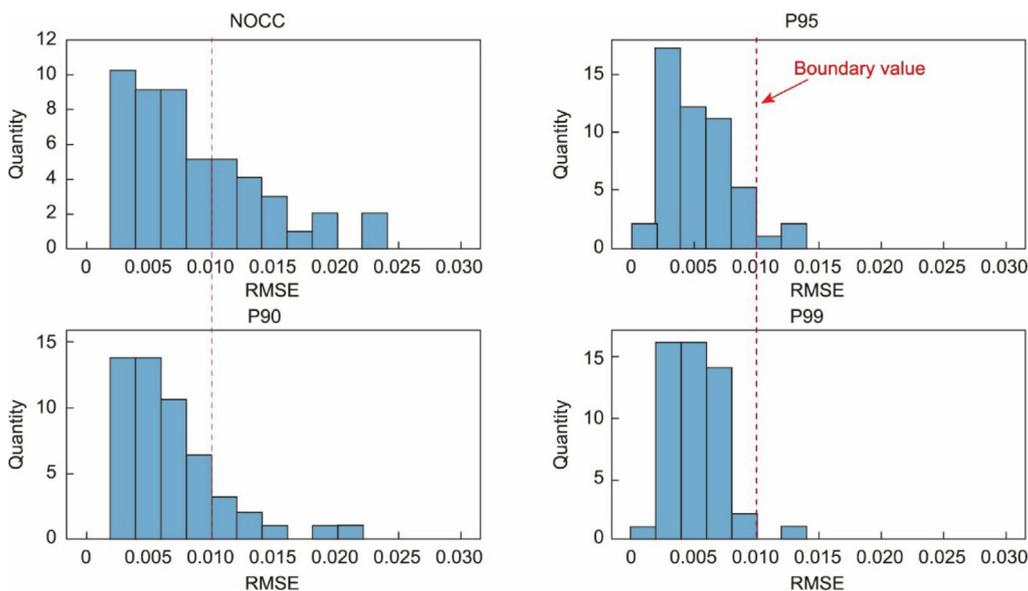


Fig. 15. Prediction performances of datasets generated using 50 normal distributed IVF values.

of the RNN models. The transition toward smaller RMSE values indicates improved accuracy in predicting the moisture content, while the narrowing of the distribution reveals a reduction in prediction variability. As shown in Figs. 17 and 18, P99 and P95 pro-

vide a narrower RMSE distribution curve shape with smaller RMSE values, which indicates reliable and consistent prediction performance for both random and distributed selected IVF values. In comparison with the CCRNN models, the RMSE predicted by

Table 7
Violation rates of Tests 2, 3, and 4.

Model	Random IVF (Test 2)		Normal distributed IVF (Test 3)		Right-skewed distributed IVF (Test 4)	
	Violations (RMSE > 0.01)	Rate	Violations (RMSE > 0.01)	Rate	Violations (RMSE > 0.01)	Rate
NOCC	16/50	32%	17/50	34%	13/50	26%
P90	5/50	10%	8/50	16%	7/50	14%
P95	3/50	6%	3/50	6%	3/50	6%
P99	2/50	4%	1/50	2%	2/50	4%

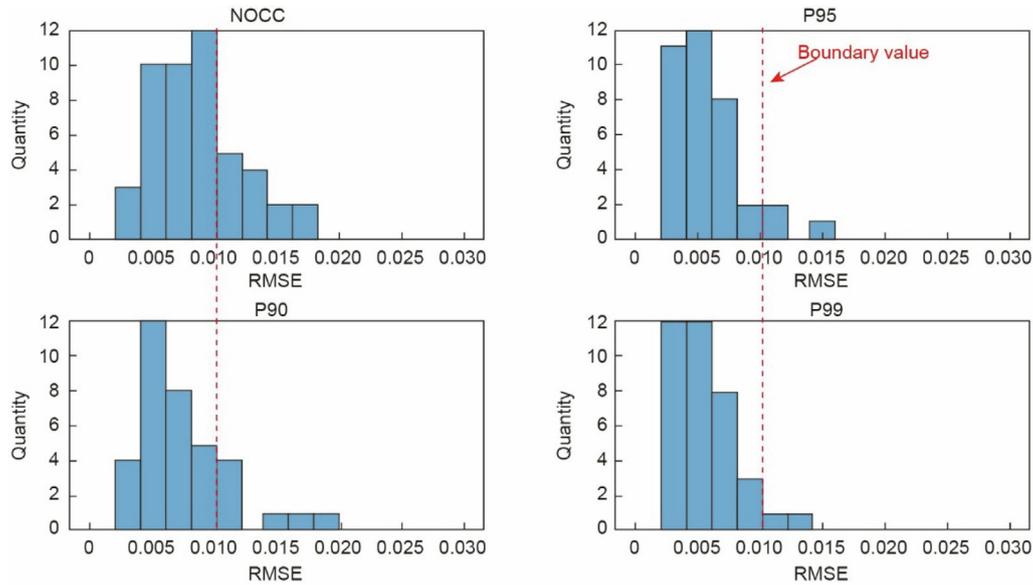


Fig. 16. Prediction performances of datasets generated using 50 right-skewed (log-normal) distributed IVF values.

NOCC is highly likely to exceed 0.01 when material variation is involved.

In summary, Tests 2 and 3 illustrate the same conclusion as Test 1: The CCRNN models demonstrate greater flexibility, resilience to practical manufacturing-similar datasets, and more accurate prediction of the CQA of moisture content compared with the general RNN model. Test 4 demonstrates that different types of IVF value distributions do not have a significant influence on moisture con-

tent prediction performance. Additionally, the RMSE violation rate in Table 7 shows a trend of satisfying the confidence level, which explains the correlation between CCRNN model prediction reliability and training dataset selection. This can lead to increased reliability and trustworthiness in the model's predictions, which can directly affect decision-making when operating a manufacturing process, reduce the risk of unexpected outcomes, and improve process control.

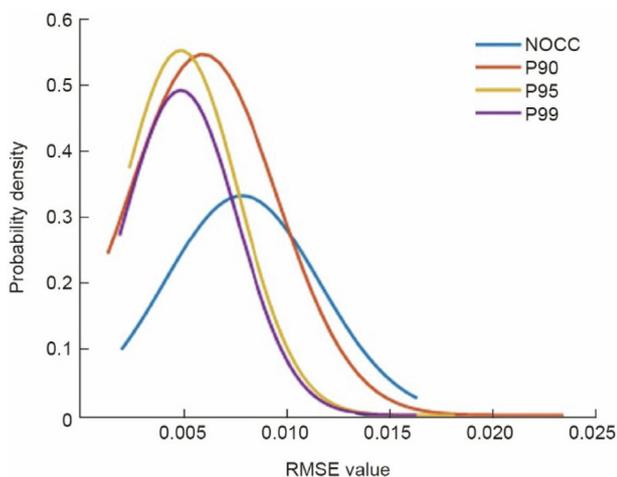


Fig. 17. RNN model prediction RMSE distribution shift using a random selection of IVF values.

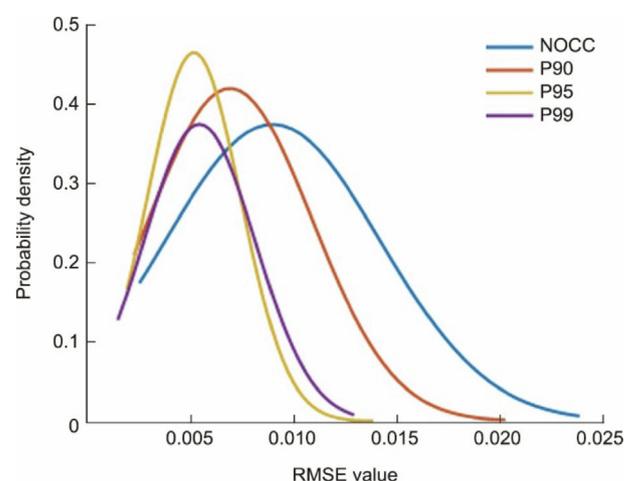


Fig. 18. RNN model prediction RMSE distribution shift using normal distributed IVF values.

5. Conclusions and future work

The aim of this research was to investigate how considering material quality uncertainty during the training process impacts the prediction accuracy for the CQA and the overall robustness of RNN models. The observed results revealed that, compared with the traditional RNN model, CCRNN models offered more robust, flexible, and accurate performance in predicting the granule moisture content of the fluid bed drying process in the presence of material quality variation. The strength of the proposed approach is that it incorporates material variation uncertainty into the RNN model training process through quantification using data-driven chance-constrained programming before data collection.

It was notable that P99 had the best performance among the three CCRNN models, with a smaller RMSE value and a narrower RMSE distribution curve, followed by P95 and P90. This observation explains the potential correlation between CCRNN model reliability and its prediction performance, where a lower risk level results in a lower violation rate. However, one limitation of this method is that the distribution information of the uncertain variable must be obtainable or known, although it does not have to be a normal distribution.

This study explored a single uncertain material variable in the training of RNN models. Future work will focus on investigating more complex scenarios associated with multiple uncertain variables, including process parameter uncertainty and uncertainty in the external environment [63,64]. Joint chance constraints could be a potential solution to deal with multiple uncertainty issues.

CRedit authorship contribution statement

Qingbo Meng: Visualization, Investigation, Methodology, Data curation, Writing – original draft, Formal analysis. **I. David L. Bogle:** Supervision, Funding acquisition, Writing – review & editing, Investigation. **Vassilis M. Charitopoulos:** Supervision, Funding acquisition, Methodology, Writing – review & editing, Investigation, Conceptualization.

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.

Acknowledgment

Financial support from the Engineering and Physical Sciences Research Council grant EP/V034723/1 (RiFTMaP) is gratefully acknowledged.

References

- [1] Ierapetritou M, Muzzio F, Reklaitis G. Perspectives on the continuous manufacturing of powder-based pharmaceutical processes. *AIChE J* 2016;62(6):1846–62.
- [2] Wang Z, Ierapetritou M. Applications of optimization in the pharmaceutical process development. In: Muzzio FJ, Oka S, editors. *How to design and implement powder-to-tablet continuous manufacturing systems*. London: Elsevier; 2022. p. 271–99.
- [3] Bogle IDL. A perspective on smart process manufacturing research challenges for process systems engineers. *Engineering* 2017;3(2):161–5.
- [4] Napoleone A, Pozzetti A, Macchi M, Andersen R. Time to be responsive in the process industry: a literature-based analysis of trends of change, solutions and challenges. *Prod Plann Contr* 2023;34(6):572–86.
- [5] Litster J, Bogle IDL. Smart process manufacturing for formulated products. *Engineering* 2019;5(6):1003–9.
- [6] Chen Y, Yang O, Sampat C, Bhalode P, Ramachandran R, Ierapetritou M. Digital twins in pharmaceutical and biopharmaceutical manufacturing: a literature review. *Processes* 2020;8(9):1088.
- [7] Sharifian S, Sotudeh-Gharebagh R, Zarghami R, Tanguy P, Mostoufi N. Uncertainty in chemical process systems engineering: a critical review. *Rev Chem Eng* 2021;37(6):687–714.
- [8] Department of Health and Human Services, US Food and Drug Administration. Pharmaceutical CGMPs for the 21st century—a risk-based approach. Report. Rockville: US Food and Drug Administration; 2004.
- [9] Boukouvala F, Ierapetritou MG. Surrogate-based optimization of expensive flowsheet modeling for continuous pharmaceutical manufacturing. *J Pharm Innov* 2013;8(2):131–45.
- [10] Boukouvala F, Muzzio FJ, Ierapetritou MG. Design space of pharmaceutical processes using data-driven-based methods. *J Pharm Innov* 2010;5(3):119–37.
- [11] Bounititsis GL, Papageorgiou LG, Charitopoulos VM. Stable optimisation-based scenario generation via game theoretic approach. *Comput Chem Eng* 2024;185:108646.
- [12] Gao J, Liu L, Dong Y, Zhang L, Zhuang Y, Du J. Stochastic programming-based mathematical model and solution strategy for chemical production scheduling with processing time uncertainty. *Comput Chem Eng* 2022;168:108043.
- [13] Zeng Z, Cremaschi S. Multistage stochastic programming models for pharmaceutical clinical trial planning. *Processes* 2017;5(4):71.
- [14] Nikzad E, Bashiri M, Oliveira F. Two-stage stochastic programming approach for the medical drug inventory routing problem under uncertainty. *Comput Ind Eng* 2019;128:358–70.
- [15] Jang HA, Kim SY, Lim YY, Lim JL, Shin S. Robust optimization approaches for a natural pharmaceutical complex product of *Atractylodes japonica* Koidz and *Schisandra chinensis*. *Appl Sci* 2020;10(19):7006.
- [16] Xie X, Schenkendorf R. Robust optimization of a pharmaceutical freeze-drying process under non-Gaussian parameter uncertainties. *Chem Eng Sci* 2019;207:805–19.
- [17] Ning C, You F. Data-driven stochastic robust optimization: general computational framework and algorithm leveraging machine learning for optimization under uncertainty in the big data era. *Comput Chem Eng* 2018;111:115–33.
- [18] Liu J, Su Q, Moreno M, Laird C, Nagy Z, Reklaitis G. Robust state estimation of feeding–blending systems in continuous pharmaceutical manufacturing. *Chem Eng Res Des* 2018;134:140–53.
- [19] Häussling Löwgren B, Weigert J, Esche E, Repke JU. Uncertainty analysis for data-driven chance-constrained optimization. *Sustainability* 2020;12(6):2450.
- [20] Calfa BA, Grossmann IE, Agarwal A, Bury SJ, Wassick JM. Data-driven individual and joint chance-constrained optimization via kernel smoothing. *Comput Chem Eng* 2015;78:51–69.
- [21] Kumar Ananthu M, Kumar Chintamaneni P, Basha Shaik S, Thadipatri R, Mahammed N. Artificial neural networks in optimization of pharmaceutical formulations. *Saudi J Med Pharm Sci* 2021;7(8):368–78.
- [22] Tu JV. Advantages and disadvantages of using artificial neural networks versus logistic regression for predicting medical outcomes. *J Clin Epidemiol* 1996;49(11):1225–31.
- [23] Wesolowski M, Suchacz B. Artificial neural networks: theoretical background and pharmaceutical applications: a review. *J AOAC Int* 2012;95(3):652–68.
- [24] Zhang Z, Zhou D, Zhang J, Xu Y, Lin G, Jin B, et al. Multilayer perceptron-based prediction of stroke mimics in prehospital triage. *Sci Rep* 2022;12(1):17994.
- [25] Khalid MH, Kazemi P, Perez-Gandarillas L, Michrafy A, Szlek J, Jachowicz R, et al. Computational intelligence models to predict porosity of tablets using minimum features. *Drug Des Devel Ther* 2017;11:193–202.
- [26] Korteby Y, Mahdi Y, Azizou A, Daoud K, Regdon Jr G. Implementation of an artificial neural network as a PAT tool for the prediction of temperature distribution within a pharmaceutical fluidized bed granulator. *Eur J Pharm Sci* 2016;88:219–32.
- [27] Han R, Yang Y, Li X, Ouyang D. Predicting oral disintegrating tablet formulations by neural network techniques. *Asian J Pharm Sci* 2018;13(4):336–42.
- [28] Galata DL, Farkas A, Könyves Z, Mészáros LA, Szabó E, Csontos I, et al. Fast, spectroscopy-based prediction of *in vitro* dissolution profile of extended release tablets using artificial neural networks. *Pharmaceutics* 2019;11(8):400.
- [29] Velásco-Mejía A, Vallejo-Becerra V, Chávez-Ramírez AU, Torres-González J, Reyes-Vidal Y, Castañeda-Zaldivar F. Modeling and optimization of a pharmaceutical crystallization process by using neural networks and genetic algorithms. *Powder Technol* 2016;292:122–8.
- [30] Madzarevic M, Medarevic D, Vulovic A, Sustersic T, Djuris J, Filipovic N, et al. Optimization and prediction of ibuprofen release from 3D DLP printlets using artificial neural networks. *Pharmaceutics* 2019;11(10):544.
- [31] Meng D, Liu Z. Machine learning aided pharmaceutical engineering: model development and validation for estimation of drug solubility in green solvent. *J Mol Liq* 2023;392(Pt 1):123286.
- [32] Ragb HK, Gopal P. Multi-layered deep learning perceptron based model for predicting drug price changes. 2023. [TechRxiv: 24417697v1](https://arxiv.org/abs/24417697v1).
- [33] Metta N, Ramachandran R, Ierapetritou M. A novel adaptive sampling based methodology for feasible region identification of compute intensive models using artificial neural network. *AIChE J* 2021;67(2):e17095.
- [34] Park YS, Lek S. Artificial neural networks: multilayer perceptron for ecological modeling. In: Jørgensen SE, editor. *Developments in environmental modelling*. Amsterdam: Elsevier; 2016. p. 123–40.

- [35] Montesinos López OA, Montesinos López A, Crossa J. Fundamentals of artificial neural networks and deep learning. In: Montesinos López OA, Montesinos López A, Crossa J, editors. *Multivariate statistical machine learning methods for genomic prediction*. Cham: Springer; 2022. p. 379–425.
- [36] Sarker IH. Deep learning: a comprehensive overview on techniques, taxonomy, applications and research directions. *SN Comput Sci* 2021;2:420.
- [37] Wong WC, Chee E, Li J, Wang X. Recurrent neural network-based model predictive control for continuous pharmaceutical manufacturing. *Mathematics* 2018;6(11):242.
- [38] Baranilingesan I. Optimization algorithm-based Elman neural network controller for continuous stirred tank reactor process model. *Curr Sci* 2021;120(8):1324–33.
- [39] Vega-Zambrano C, Diangelakis NA, Charitopoulos VM. Data-driven model predictive control for continuous pharmaceutical manufacturing. *Int J Pharmaceutics* 2025;672:125322.
- [40] Vora LK, Gholap AD, Jetha K, Thakur RRS, Solanki HK, Chavda VP. Artificial intelligence in pharmaceutical technology and drug delivery design. *Pharmaceutics* 2023;15(7):1916.
- [41] Wang S, Di J, Wang D, Dai X, Hua Y, Gao X, et al. State-of-the-art review of artificial neural networks to predict, characterize and optimize pharmaceutical formulation. *Pharmaceutics* 2022;14(1):183.
- [42] Nagy B, Galata DL, Farkas A, Nagy ZK. Application of artificial neural networks in the process analytical technology of pharmaceutical manufacturing—a review. *AAPS J* 2022;24:74.
- [43] Peng J, Li J, Shang X. A learning-based method for drug–target interaction prediction based on feature representation learning and deep neural network. *BMC Bioinf* 2020;21(S13):394.
- [44] Bizmark N, Mostoufi N, Sotudeh-Gharebagh R, Ehsani H. Sequential modeling of fluidized bed paddy dryer. *J Food Eng* 2010;101(3):303–8.
- [45] Izadifar M, Mowla D. Simulation of a cross-flow continuous fluidized bed dryer for paddy rice. *J Food Eng* 2003;58(4):325–9.
- [46] Mohseni M, Kolomijtschuk A, Peters B, Demoulling M. Biomass drying in a vibrating fluidized bed dryer with a Lagrangian–Eulerian approach. *Int J Therm Sci* 2019;138:219–34.
- [47] Handayani SU, Yohana E, Tauviqirrahman M, Rahman AG, Yulianto ME, Choi KH. Performance improvement of continuous horizontal fluidised bed dryer based on computational fluid dynamics. *Results Eng* 2023;17:100972.
- [48] Burgschweiger J, Tsotsas E. Experimental investigation and modelling of continuous fluidized bed drying under steady-state and dynamic conditions. *Chem Eng Sci* 2002;57(24):5021–38.
- [49] Wang LG, Morrissey JP, Barrasso D, Slade D, Clifford S, Reynolds G, et al. Model driven design for twin screw granulation using mechanistic-based population balance model. *Int J Pharm* 2021;607:120939.
- [50] Jiang SL, Papageorgiou LG, Bogle IDL, Charitopoulos VM. Investigating the trade-off between design and operational flexibility in continuous manufacturing of pharmaceutical tablets: a case study of the fluid bed dryer. *Processes* 2022;10(3):454.
- [51] Charitopoulos VM, Aguirre AM, Papageorgiou LG, Dua V. Uncertainty aware integration of planning, scheduling and multi-parametric control. In: Eden MR, Ierapetritou MG, Towler GP, editors. *Computer aided chemical engineering*. Amsterdam: Elsevier; 2018. p. 1171–6.
- [52] Jiang R, Guan Y. Data-driven chance constrained stochastic program. *Math Program* 2016;158(1–2):291–327.
- [53] Li P, Arellano-Garcia H, Wozny G. Chance constrained programming approach to process optimization under uncertainty. *Comput Chem Eng* 2008;32(1–2):25–45.
- [54] Pascanu R, Mikolov T, Bengio Y. On the difficulty of training recurrent neural networks. In: Dasgupta S, McAllester D, editors. *Proceedings of the 30th International Conference on International Conference on Machine Learning*; 2013 Jun 16–21; Atlanta, GA, USA. New York City: PMLR; 2012. p. 1310–8.
- [55] Bengio Y, Simard P, Frasconi P. Learning long-term dependencies with gradient descent is difficult. *IEEE Trans Neural Netw* 1994;5(2):157–66.
- [56] Zhang J, Zhu Y, Zhang X, Ye M, Yang J. Developing a long short-term memory (LSTM) based model for predicting water table depth in agricultural areas. *J Hydrol* 2018;561:918–29.
- [57] Yang SB, Li Z. Recurrent neural network-based joint chance constrained stochastic model predictive control. *IFAC-PapersOnLine* 2022;55(7):780–5.
- [58] Sagheer A, Kotb M. Time series forecasting of petroleum production using deep LSTM recurrent networks. *Neurocomputing* 2019;323:203–13.
- [59] Shah SRB, Chadha GS, Schwung A, Ding SX. A sequence-to-sequence approach for remaining useful lifetime estimation using attention-augmented bidirectional LSTM. *Intell Syst Appl* 2021;10–11:200049.
- [60] Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R. Dropout: a simple way to prevent neural networks from overfitting. *J Mach Learn Res* 2014;15(1):1929–58.
- [61] Piotrowski AP, Napiorkowski JJ, Piotrowska AE. Impact of deep learning-based dropout on shallow neural networks applied to stream temperature modelling. *Earth Sci Rev* 2020;201:103076.
- [62] Meng Q, Bogle IDL, Charitopoulos VM. Data-driven chance-constrained optimization for minimizing the influence of material uncertainty on product quality. In: Manenti F, Reklaitis GV, editors. *Computer aided chemical engineering*. Amsterdam: Elsevier; 2024. p. 1579–84.
- [63] Charitopoulos VM, Papageorgiou LG, Dua V. Multi set-point explicit model predictive control for nonlinear process systems. *Processes* 2021;9(7):1156.
- [64] Charitopoulos VM, Dua V. Explicit model predictive control of hybrid systems and multiparametric mixed integer polynomial programming. *AIChE J* 2016;62(9):3441–60.