Deep learning for kinetics parameters identification: A novel approach for multi-variate optimization

Chemical Engineering Journal 489 (2024) 151149

Contents lists available at ScienceDirect

Chemical Engineering Journal

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

Deep learning for kinetics parameters identification: A novel approach for multi-variate optimization

Antonello Raponi *, Daniele Marchisio

Politecnico di Torino, Institute of Chemical Engineering, Department of Applied Science and Technology, 10129 Torino, Italy

ARTICLE INFO

Keywords:
Machine learning
Population balance model
Precipitation
Predictive framework
Optimization

ABSTRACT

This work presents an innovative deep-learning approach for multi-variate optimization, focusing on the identification of $Mg(OH)_2$ precipitation kinetics parameters. The study employs three distinct experimental datasets, one for the Population Balance Model (PBM) fitting and two for validation. These datasets explore the impact on Particle Size Distributions (PSDs) of (i) increasing the initial reactant concentrations from 0.125 to 1 M and (ii) decreasing the flow rate from 12 to 4 m/s, both in a T-mixer, (iii) increasing the initial reactant concentration over a wider concentration range from 0.01 to 1 M in a more complex Y-mixer system. Leveraging PBM, we create a dataset to train a Neural Network (NN), referred to as the 'mirror model,' which predicts kinetics parameters based on experimental sizes. Notably, the PBM, fitted with dataset (i), excels at describing changes in flow rate (dataset (ii)) and substantial reductions in reactant concentrations in the Y-mixer (dataset (iii)), even though these conditions were not encountered during the fitting step. Key Performance Indicators (KPIs) reveal that the mirror model consistently outperforms two widely used algorithms, Conjugate Gradient (CG) and Particle Swarm Optimization (PSO), highlighting its remarkable potential for practical applications.

1. Introduction

In recent years, the list of Critical Raw Materials (CRMs) has grown, with magnesium being included in this category [1,2]. A highly effective method for magnesium recovery involves using it as a precursor to synthesize magnesium hydroxide ($Mg(OH)_2$), which finds applications across numerous industries [3–9]. $Mg(OH)_2$ serves as a flame-retardant filler, offering fire-retardant properties through endothermic dehydration at elevated temperatures. It also excels as an efficient acidic waste neutralizer, aligning with environmentally friendly practices. Within the pharmaceutical sector, $Mg(OH)_2$ acts as an excipient, playing an important role in various drug formulations. Additionally, it extends its utility to the pulp and paper sector, where it serves as a preservative, enhancing the quality and longevity of paper products. The traditional method for $Mg(OH)_2$ synthesis has been the hydrothermal route [10]. However, this route comes with the primary drawback of a high energy demand due to the elevated temperature requirements. Consequently, there has been a shift towards precipitation as a more cost-effective and sustainable synthesis route [11].

The precipitation process involves using magnesium sources such as brines or bitterns, which react with an alkaline solution [12]. As a consequence of the reaction, $Mg(OH)_2$ is formed and remains in solution as long as the concentration remains below the solubility limit. When the concentration exceeds this limit, $Mg(OH)_2$ precipitates forming particles of different sizes. Precipitation can be, therefore, described through a Population Balance Model (PBM) [13,14] that can track the evolution of the particle size distribution (PSD). Multiple phenomena influence the PSD, including primary nucleation, molecular growth, and aggregation. Therefore, the identification of precipitation kinetics becomes crucial in optimizing the PSD required for each application [15,16]. In line with numerous chemical engineering processes [17–19], the precipitation kinetics set can be identified by comparison between the PBM outcomes and experimental data through an optimization algorithm. The effectiveness of optimization algorithms has grown over the years, driven by the increasing computational power available. These advancements are particularly significant in various engineering domains, including process engineering, where researchers frequently encounter large-scale multi-variate problems (LSMVPs) [20]. LSMVPs are typically characterized by extensive search spaces which classical optimization algorithms, such as the conjugate gradient (CG), might not be able to fully explore. An alternative approach involves modifying the CG method by tailoring the line search [21,22]. While these modified methods are more robust and reliable, they may still be inefficient for multi-variate optimizations [23], where the optimized solution strongly depends on the first-attempt value. A viable alternative that has found

* Corresponding author.
E-mail address: antonello.raponi@polito.it (A. Raponi).

https://doi.org/10.1016/j.cej.2024.151149
Received 12 January 2024; Received in revised form 14 March 2024; Accepted 8 April 2024
Available online 9 April 2024

widespread use in recent years is the employment of heuristic [24] and meta-heuristic methods [25]. Heuristic optimization is a method designed to discover the optimal solution to a problem within a search space that is too extensive to be thoroughly explored using conventional approaches [26]. Meta-heuristic optimization is a higher-level procedure that aims to find, select, and calibrate a heuristic algorithm that can provide the optimal solution for a specific LSMVP [27]. The former includes Particle Swarm Optimization (PSO) [28], while the latter includes Crow Search Optimization (CSO) [29]. These methods offer significant improvements over classical optimization but may require a high number of iterations for precise solutions. A promising compromise between accuracy and computational time involves leveraging artificial intelligence [30]. Hybrid models that integrate first principles models with experimental data through artificial intelligence have shown effectiveness in recent studies [31]. These models combine mechanistic equations that describe physical phenomena with experimental evidence to create a comprehensive understanding of the system.

In this contribution, a PBM is employed along with three experimental datasets (one for PBM fitting and two for validation) from previous publications of ours [32,33]. The PBM describes particle size evolution due to primary nucleation (homogeneous and heterogeneous), molecular growth, and irreversible aggregation resulting from cementation between particles. The PBM is solved through the Quadrature Method of Moments (QMOM) [34,35]. It generates numerical data for training a deep-learning fully connected neural network (NN), which is guided by experimental data to identify unknown PBM parameters.

The manuscript is organized as follows: in Section 2, we provide an overview of the experimental protocols and datasets gathered from both T- and Y-mixer systems, covering diverse operating conditions. Section 3 offers a presentation of the PBM, with a specific focus on its inputs and outputs. Our novel deep learning-assisted methodology is detailed in Section 4. Lastly, Section 5 presents the results and conducts a comparative analysis between the NN and CG and PSO performances. Conclusions are drawn in Section 6.

2. Methodology

To provide the necessary context, a brief background on the used experimental rigs and collected experimental data is presented.

Experimental datasets. In our previous works [32,33], we developed an experimental protocol to characterize $Mg(OH)_2$ suspensions obtained under various operating conditions in different systems. This protocol was, therefore, applied to suspensions obtained through two experimental apparatuses, each equipped with a high-mixing efficiency device: (i) a T-mixer and (ii) a Y-mixer, followed by two divergent channel sections and a final coil with a constant diameter, as illustrated in Fig. 1.

Although the experimental protocol is extensively detailed in the aforementioned publications, the key steps are reported here. Magnesium chloride ($MgCl_2$) and sodium hydroxide (NaOH) solutions were fed into the mixers' inlets. The solutions reacted in the mixing channel leading to the formation of $Mg(OH)_2$ suspension following the reaction:

$$Mg^{2+}_{(aq)} + 2OH^{-}_{(aq)} \rightarrow Mg(OH)_{2(s)} \downarrow \tag{1}$$

The suspension was collected, stabilized, and analyzed using the Dynamic Light Scattering (DLS) technique to obtain the PSDs. From each PSD, the jth order moment, m_j, can be calculated as follows:

$$m_j = \int_0^\infty L^j f(L) dL \tag{2}$$

Here, L is the particle size and $f(L)$ is the PSD. Once the moments are calculated, characteristic sizes, defined by the ratio of consecutive moments, can be obtained:

$$d_{i,j} = \frac{m_i}{m_j}, (i = j + 1) \tag{3}$$

This protocol aimed to evaluate the impact of various process variables on the PSDs, including initial reactant concentrations and flow rates. The investigated operating conditions in both mixers are detailed in Table 1.

Two sets of experiments were conducted in the T-mixer: #1 focused on increasing the concentration of Mg^{2+} from 0.125 to 1 M at a constant flow rate of 2320 ml/min in the T-mixer, while #2 studied the effect of decreasing flow rate to 1602 and 773 mL/min in the T-mixer at a constant concentration of Mg^{2+} of 1 M. Experiments in #3 explored variations in the concentration of Mg^{2+} over a wider range from 0.01 to 1 M at a constant flow rate of 835 mL/min in the Y-mixer. The Y-mixer's design allowed for the investigation of lower concentrations due to the final pipe, which could be extended to ensure sufficiently long residence times for the reaction completion. In Table 1 the Reynolds number and flow rate refer to the mixing channel, the NaOH concentration is kept at the stoichiometric ratio (1:2) for each experiment and the mixing time was estimated using Eq. (6) in Schikarski et al. [36]. Four characteristic sizes (d_{10}, d_{21}, d_{32}, d_{43}) are experimentally derived for each operating condition and used in numerical optimization. Dataset #1 (20 points) is used for fitting the PBM parameters, whereas datasets #2 and #3 (32 points) are used for testing. For the sake of clarity, the discussion primarily focuses on d_{10} as the other characteristic sizes exhibit a similar qualitative trend but with higher values ($d_{10} < d_{21} < d_{32} < d_{43}$) [33]. The d_{10} values for each dataset are presented in Fig. 2.

3. Population balance model

To enhance clarity and accessibility, a concise description of the PBM used is provided, focusing on the model's inputs and outputs. The PBM is mono-dimensional (1D) and assumes that the system behaves like a Plug-Flow Reactor (PFR). This model uses Computational Fluid Dynamics (CFD) simulations to incorporate information about the flow field and turbulence. Specifically, it relies on Reynolds-Averaged Navier–Stokes (RANS) simulations. The micro-mixing is addressed by tracking the evolution of the mixture fraction variance, which is incorporated into the PBM using the β-PDF approach. The model solves the Population Balance Equation (PBE), via the Quadrature Method of Moments (QMOM), returning as outputs the characteristic particle sizes (d_{10}, d_{21}, d_{32}, d_{43}) in terms of moment ratios and these characteristic sizes can be compared with those provided by the experimental tests. $Mg(OH)_2$ precipitation takes place when Mg^{2+} and OH^- ions reach the molecular scale and react (Eq. (1)) generating supersaturation (S):

$$S = \frac{\gamma_{\pm}^3 \left(\bar{c}_{Mg^{2+}} \bar{c}^2_{OH^-} \right) - k_{sp}}{k_{sp}} \tag{4}$$

Here, $\gamma_{\pm}$ is the multi-component solution activity coefficient, $\bar{c}_{Mg^{2+}}$ and $\bar{c}_{OH^-}$ are the concentrations of ions reacting as a result of their micro-mixing and k_{sp} is the solubility product. The supersaturation generation triggers the nucleation of nuclei, which can then grow ('primary particles') and aggregate, leading to the formation of bigger clusters ('secondary particles' or, shortly, 'particles'). The processes of primary nucleation, molecular growth, and aggregation can be effectively modeled using constitutive equations, also known as kernels. In our model, we employed the Volmer-Weber equation to describe the nucleation rate (J), accounting for both the homogeneous and heterogeneous contributions:

$$J = \underbrace{A_1 e^{\left(-\frac{B_1}{\ln(S+1)^2}\right)}}_{\text{Homogeneous}} + \underbrace{A_2 e^{\left(-\frac{B_2}{\ln(S+1)^2}\right)}}_{\text{Heterogeneous}} \tag{5}$$

Here, A_1 and A_2 $\left(\text{particle no. } m^{-3}s^{-1}\right)$ are parameters describing the nucleation rate at infinite supersaturation ($S \rightarrow \infty$) and B_1 and B_2 (dimensionless) describe the interaction between the liquid and the $Mg(OH)_2$ particles surface or foreign solids surface, respectively. For the molecular growth rate (G), a power law expression was chosen:

$$G = k_g S^g \tag{6}$$

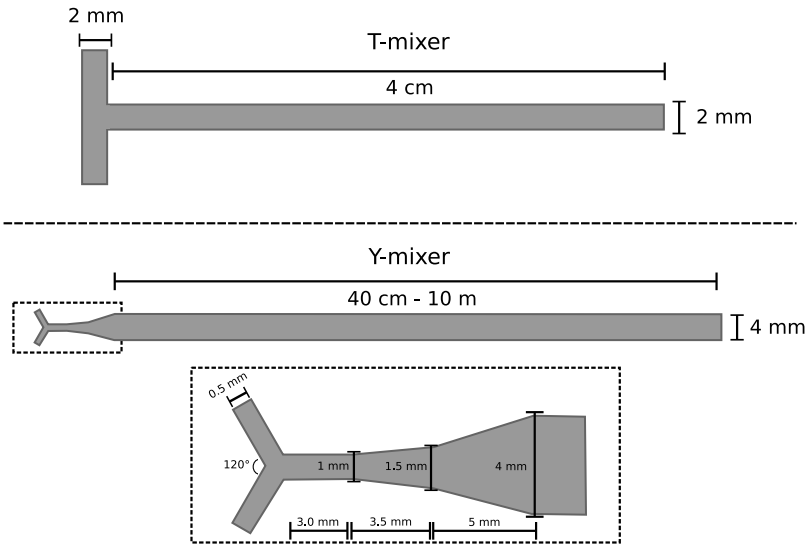


Fig. 1. Experimental setups: T-mixer (*top*) and Y-mixer (*bottom*).

Table 1
Investigated operating conditions for the T- and Y-mixer.

Dataset	Geometry	Operating conditions				
		Concentration (M)	Residence time (ms)	Reynolds number ($\cdot 10^4$)	Flow rate (mL/min)	Estimate mixing time (ms)
#1	T-Mixer	0.125	~ 3	2.7	2320	2
		0.25	~ 3	2.7	2320	2
		0.5	~ 3	2.7	2320	2
		0.75	~ 3	2.7	2320	2
		1	~ 3	2.7	2320	2
#2	T-Mixer	1	~ 5	1.7	1602	2.8
		1	~ 10	0.8	773	5.9
#3	Y-Mixer	0.01	$\sim 9 \cdot 10^3$	1.7	835	0.6
		0.025	$\sim 9 \cdot 10^3$	1.7	835	0.6
		0.05	$\sim 9 \cdot 10^3$	1.7	835	0.6
		0.125	$\sim 3.6 \cdot 10^2$	1.7	835	0.6
		0.5	$\sim 3.6 \cdot 10^2$	1.7	835	0.6
		1	$\sim 3.6 \cdot 10^2$	1.7	835	0.6

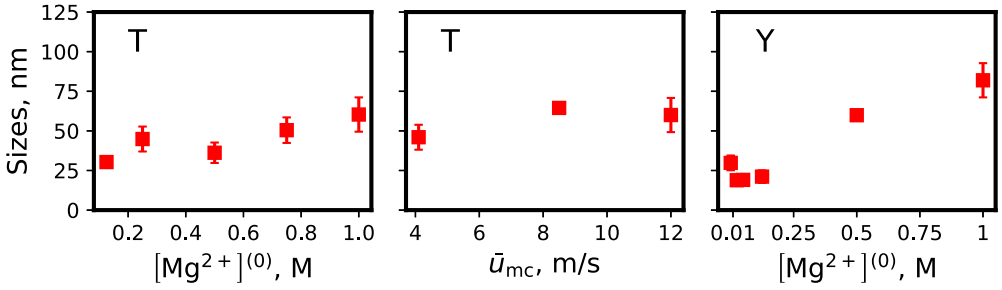


Fig. 2. Experimental d_{10} trends: dataset #1 (*left*), dataset #2 (*center*), dataset #3 (*right*).

Here, k_g (ms^{-1}) is a parameter describing the rate at which particles grow while g (dimensionless) describes the mechanism by which growth occurs. Furthermore, the aggregation rate (β_{agg}) was described as the product of collision frequency (β_{coll}) and sticking probability (ψ), using the following equation:

$$\beta_{\mathrm{agg}} = \beta_{\mathrm{coll}}\psi \tag{7}$$

We assumed that the collision frequency depended on both turbulent fluctuations (β^{tr}) and thermal, or Brownian, fluctuations (β^{br}), while the sticking probability was a function of the cementation time (t_c) and the interaction time (t_i):

$$\beta_{\mathrm{coll}} = 10^{C_1}\left(\beta^{\mathrm{tr}} + \beta^{\mathrm{br}}\right) \tag{8}$$

$$\beta^{\mathrm{tr}} = \sqrt{\frac{8\pi}{15}}\sqrt{\frac{\varepsilon}{\nu}}\frac{(L+\lambda)^3}{8} \tag{9}$$

$$\beta^{\mathrm{br}} = \frac{2k_{\mathrm{B}}T}{3\mu}\frac{(L+\lambda)^2}{L\lambda} \tag{10}$$

$$\psi(A_{\mathrm{P}},\varepsilon) = \mathrm{e}^{-\frac{t_c(A_{\mathrm{P}},\varepsilon)}{t_i(\varepsilon)}} \tag{11}$$

Here, C_1 (dimensionless) is a correction factor added to the equation to account for any deviation from ideality, while A_{p} (Nm^{-2}) is proportional to the strength of the crystalline bridge formed between the particles, ϵ is the dissipation of turbulent kinetic energy, ν is the kinematic viscosity of the fluid, L and λ are the sizes of the colliding particles, k_{B} is the Boltzmann constant, T the fluid temperature, μ the

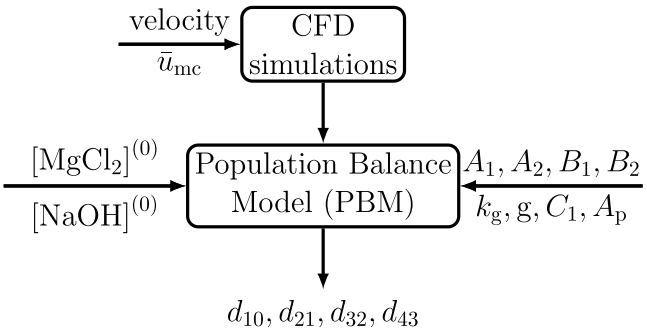


Fig. 3. Model flowchart.

dynamic viscosity of the fluid. Cementation and interaction time, on the other hand, are expressed as:

$$t_c = \frac{D_b}{f_s G} \tag{12}$$

$$t_i = \sqrt{\frac{\nu}{\epsilon}} \tag{13}$$

$$D_b = \frac{L_{eq}\rho_c^{0.5}(\epsilon\nu)^{0.25}}{A_p^{0.5}} \tag{14}$$

$$L_{eq} = \frac{L\lambda}{\left(L^2+\lambda^2-L\lambda\right)^{0.5}} \tag{15}$$

Here, f_s is a shape function [37] and ρ_c is the crystal density. In total, then, eight parameters need to be identified: four from the primary nucleation (A_1, A_2, B_1, B_2), two from the molecular growth (k_g, g) and two from the aggregation rate equations (C_1, A_p). The values of these parameters dictate the magnitude of source terms in the equations that describe the moments' evolution:

$$\frac{d\left(\bar{u}m_j\right)}{dy} = L_c^j J + \int_0^\infty j L^{j-1} G f dL + \bar{B}_j - \bar{D}_j \tag{16}$$

Here, L_c represents the critical size of stable nuclei (1 nm) whereas $\bar{B}_j$ and $\bar{D}_j$ address the birth and death contribution due to aggregation.

Fig. 3 presents the flowchart of the PBM framework, with a focus on inputs and outputs. Initially, CFD simulations are conducted for a specified flow rate in the mixing channel, corresponding to a velocity $\bar{u}_{mc}$, and turbulent fields (i.e. the turbulent kinetic energy, k, and the turbulent dissipation rate, ε) are integrated into the PBM. Operating conditions, including $MgCl_2$ and NaOH initial inlet concentrations, are set. Different sizes (d_{10}, d_{21}, d_{32}, d_{43}) are obtained as PBM output depending on the kinetics parameter set. It is essential to highlight that kinetics parameters are not free to vary but must adhere to physical constraints, which can be established through references to the scientific literature [38]. For instance, in the case of poorly soluble compounds such as $Mg(OH)_2$ or barium sulfate ($BaSO_4$) [39], the literature presents clear guidelines. Karpiński and Bałdyga [40] reported for these systems a solution particle density $(\text{particle no. m}^{-3})$ ranging from 10^{17} to 10^{22} from which the lower and upper bounds for parameter A_1 can be derived accordingly. In line with Classical Nucleation Theory (CNT), A_1 and A_2 can vary by several orders of magnitude. Once the range for A_1 is chosen, the range for A_2 follows. The range values of B_1 and B_2, on the other hand, can be chosen knowing some physical properties of the system such as the interfacial tension [41]. The growth rate, whose intensity is modulated by k_g, cannot be higher than it would be in the diffusion-controlled regime and g varies between 1 and 2 depending on the mechanism by which the particles grow (diffusion- or surface integration-controlled respectively) [42]. Ultimately, C_1 corrects any deviation from ideality (namely $C_1 = 0$) up to a correction of one order of magnitude (namely $C_1 = 1$) [13,32], while A_p range is tabulated [38]. For each parameter, the lower and upper limits are given in the Table 2:

Table 2
Parameter constraints and units.

	A_1	A_2	B_1	B_2	k_g	g	C_1	A_p
Lower bound (lb)	10^{19}	10^{10}	200	10	10^{-15}	1	0	10^0
Upper bound (ub)	10^{29}	10^{18}	400	10^2	10^{-9}	2	1	10^7
Units	particle no. / $m^3 s^1$				m / s			N / m^2

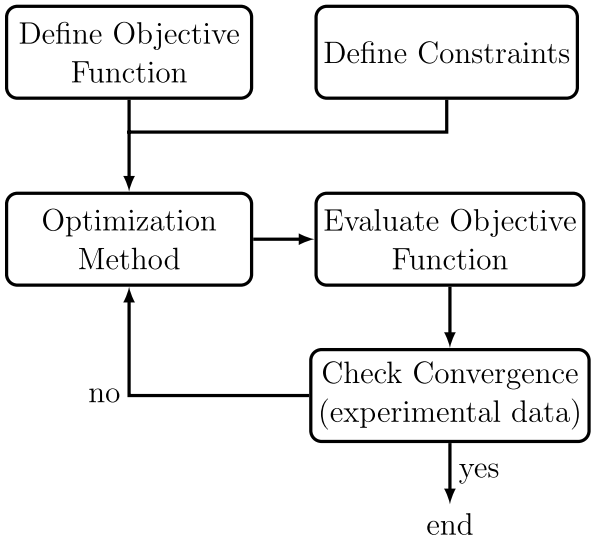


Fig. 4. Traditional optimization loop scheme.

PBM computational details. The PBM is a mono-dimensional framework coded in MATLAB, with the system of differential equations (Eq. (16)) solved using 'ode15s'. To capture the fluid dynamics within both the T-mixer and Y-mixer, OpenFOAM simulations were conducted, followed by the integration of turbulent property profiles into the mono-dimensional model. For more comprehensive insights, readers are directed to Raponi et al. [32,33].

4. Parameters identification

Parameter identification through optimization loops is the initial step in fine-tuning model parameters for process description. In this case, it is applied to determine the parameters that describe the precipitation of $Mg(OH)_2$. Generally, the traditional optimization loop can be represented as shown in Fig. 4:

The traditional optimization loop aims to optimize the model's outputs by comparison with experimental data. The PBM, for instance, takes as inputs the kinetics parameters (i.e., the control variables). These are identified by comparing the PBM outputs (i.e., the target variables) with experimental sizes at various operating conditions. Let $\vec{\varphi}$ be the vector representing the eight unknown parameters $\left(\vec{\varphi} = \varphi\left(A_1, A_2, B_1, B_2, k_g, g, C_1, A_p\right)\right)$, and $\vec{d}$ be the vector containing the model's outputs $\left(\vec{d} = d\left(d_{10}, d_{21}, d_{32}, d_{43}\right)\right)$, namely the four characteristic sizes. After implementing the model and identifying the parameters to be tuned, physical limits (i.e., the search space) are defined for each parameter (see Table 2). The core of the optimization loop is the optimization algorithm. Various algorithms are available, depending on the task and problem size, namely the number of parameters to be identified. Once the optimization algorithm is chosen, it is initialized with a first-attempt value $\left(\vec{\varphi}^I\right.$, either random or user − defined$)$. The PBM uses this value to calculate a first-attempt output $\left(\vec{d}^I\right)$, which is compared with experimental data. The goodness of the PBM output is quantified through an error that settles the distance from the experimental values.

Since the first-attempt values often return the wrong solution, the optimization algorithm iteratively adjusts the input parameters until the model and experimental data (ideally) align. However, there are limitations to traditional optimization methods, despite their robustness. Firstly, the choice of the initial first-attempt value can heavily influence the optimization process, particularly when expert guidance is lacking. Algorithms like the CG method are susceptible to this limitation. An alternative approach to address this issue is to assume a population of first-attempt values, as in the case of the PSO. Yet, this increases computational costs as the PBM must be evaluated for a larger number of points. The second limitation pertains to the definition of the error, which is user-dependent. Moreover, traditional optimization algorithms can lead to a local minimum, necessitating a large number of runs to explore the entire search space. These algorithms require simplified models (e.g., mono-dimensional) that can quickly return outputs when they are called. Lastly, an important consideration in optimization algorithms is the possibility that different combinations of parameters may lead to the same outcome. When parameter correlations are present, additional techniques such as sensitivity analysis or parameter screening may be required to identify the most influential parameters and streamline the parameter space. However, in our study, we did not encounter redundancy in outputs; consistent results were obtained across variations in the kinetics parameter sets.

Deep learning aided optimization: the mirror model. In this paragraph, we introduce a novel methodology for multi-variate optimization that leverages NNs. This innovative approach addresses the limitations discussed earlier and substantially reduces computational costs. The initial step involves creating a suitable numerical dataset using the PBM, which takes eight kinetics parameters ($\overline{\varphi}$) and the concentration of Mg^{2+} as inputs (Fig. 3). For the simulations, kinetics parameter values $\left(\overline{\varphi}^{(i)}\right)$ were randomly selected within their physical variability range, as outlined in Table 2 (for further details, see 'supporting information'). Therefore, a large search space could be covered by the PBM. Since it was used for fitting, the five concentrations from experimental dataset #1 were used in the simulations. The primary objective of the NN is to replace the whole optimization loop by creating a surrogate optimization algorithm capable of identifying precipitation kinetics (NN outputs) starting from experimental sizes (NN inputs). This NN, which mirrors the PBM inputs and outputs, is referred to as the 'mirror model' and represents a data-driven optimization algorithm. Therefore, the mirror model was trained using the four characteristic sizes at an initial Mg^{2+} concentration (five inputs) from simulations run with kinetics parameters that now serve as the mirror model outputs (eight outputs). The mirror model is schematically represented in Fig. 5.

Mirror model computational details. In this study, we conducted a preliminary investigation to determine the optimal size of the numerical dataset required for the effective generalization of patterns within the mirror model. We explored dataset sizes ranging from 100 to 400 simulations, ultimately finding that 200 simulations were sufficient. Out of these, 160 simulations were allocated for training the mirror model, with the remaining 40 reserved for testing purposes. For each dataset size, we evaluated four different neural network architectures, varying in the number of layers and neurons. Specifically, architectures included (i) 32-32, (ii) 32-64-32, (iii) 64-128-64, and (iv) 64-128-128-64 configurations. Following architectural evaluations, we assessed the impact of learning rate variation across two orders of magnitude (from 10^{-5} to 10^{-3}). The chosen mirror model architecture comprised three hidden layers with 64, 128, and 64 neurons, respectively, utilizing a learning rate of 10^{-4}. To mitigate overfitting, we employed the 'early-stopping' method. We utilized a fully connected deep learning model with Rectified Linear Unit (ReLU) activation functions for neurons in the input and hidden layers, and a linear activation function for the output layer and the Adam optimizer. The training was performed in Python utilizing TensorFlow and Keras libraries. Details regarding the mirror model structure and training/testing process can be found in the 'Supporting Information'.

5. Results and discussion

The mirror model serves as a valuable tool for identifying kinetics parameters. It operates as a data-driven model, leveraging experimental data. To illustrate its utility, let us assume to provide the four experimental characteristic sizes at a concentration of 0.125 M (corresponding to the first experimental condition in dataset #1, as detailed in Table 1) as inputs to the trained mirror model. In response, the mirror model will generate a kinetic parameter set as output. By repeating this process for the following four concentrations in dataset #1 (i.e., 0.25, 0.5, 0.75, and 1 M), four additional $\overline{\varphi}$ predictions are obtained. The five vectors $\overline{\varphi}$ obtained using the data-driven mirror model are ideally expected to be identical since the precipitation kinetics are solely related to the compound ($Mg(OH)_2$). However, in practice, achieving a perfect identity is challenging due to various sources of error, both experimental and model-related. The first category includes errors, which relate to (i) the device used for characterization, (ii) the method used for quenching to stabilize the suspension, and (iii) the time elapsed between particle synthesis and their stabilization. The second category encompasses model-related errors, which consist of (i) intrinsic numerical errors, (ii) assumptions and simplifications made during the modeling process, and (iii) stochasticity introduced during the neural network training. Considering these factors, it becomes evident that the five parameter vectors $\overline{\varphi}$ obtained using the data-driven mirror model will not be identical but will exhibit slight variations influenced by the error magnitudes. Nevertheless, this novel methodology provides a way to quantify this deviation. During the tuning phase of the kinetics set, the mirror model was queried a number of times equivalent to the number of operating conditions belonging to the fitting dataset (i.e., five in the case of dataset #1). Consequently, five sets of slightly different kinetics were obtained. For the ith parameter, therefore, a mean value ($\bar{\varphi}_i$) and a standard deviation ($\bar{\varphi}_i'^2$) have been calculated starting from these five values. The average values, denoted concisely as $\overline{\bar{\varphi}}$, serve as a measure to evaluate the overall average performance of the PBM. On the other hand, the confidence interval for precipitation kinetics, proportional to the standard deviation for each parameter $\overline{\bar{\varphi}'^2}$, is employed to quantify the uncertainty associated with the PBM trend. Let us assume, therefore, to run a statistically significant number of PBM simulations for each concentration (see 'supporting information'). These PBM simulations have inputs in the form of vectors $\overline{\varphi}$ chosen as follows: a random vector that falls within the range between $\overline{\bar{\varphi}} - \overline{\bar{\varphi}'^2}$ and $\overline{\bar{\varphi}} + \overline{\bar{\varphi}'^2}$ is selected. This procedure allows for quantifying, through the PBM, the average trend and the confidence interval as the operating conditions vary. The kinetics set and its confidence interval is, then, tested for datasets #2 and #3 as shown in Fig. 6:

While we have previously discussed average trends in two of our prior publications [32,33], the proposed methodology allows for a more detailed analysis. Indeed, the combined use of the mirror model and PBM make it possible to understand, through the solved equations, the interaction among the phenomena involved. To begin, we briefly outline the interpretation of average trends. Comparing the PBM outcomes for dataset #1 (Fig. 6-*left*), we observe that in the T-mixer there is a monotonically increasing trend in particle size as the concentration increases. On the other hand, when we analyze the PBM trend with dataset #2 (Fig. 6-*center*), we find no significant changes in particle sizes with decreasing flow rate within the mixing channel (or equivalently, as $\bar{u}_{mc}$ decreases). Finally, when the system transitions from a simple T-mixer to a more complex Y-mixer with diverging channels and a final pipe, the PBM continues to accurately predict the particle size trend, including the minimum (Fig. 6-*right*). This shifting from the T- to the Y-mixer is accounted for in the PBM by running additional CFD simulations to characterize the new flow and turbulent fields (see Fig. 3, 'CFD simulations' box). It is worth emphasizing that the PBM consistently and correctly predicted particle size trends under entirely different conditions, even though these conditions were not

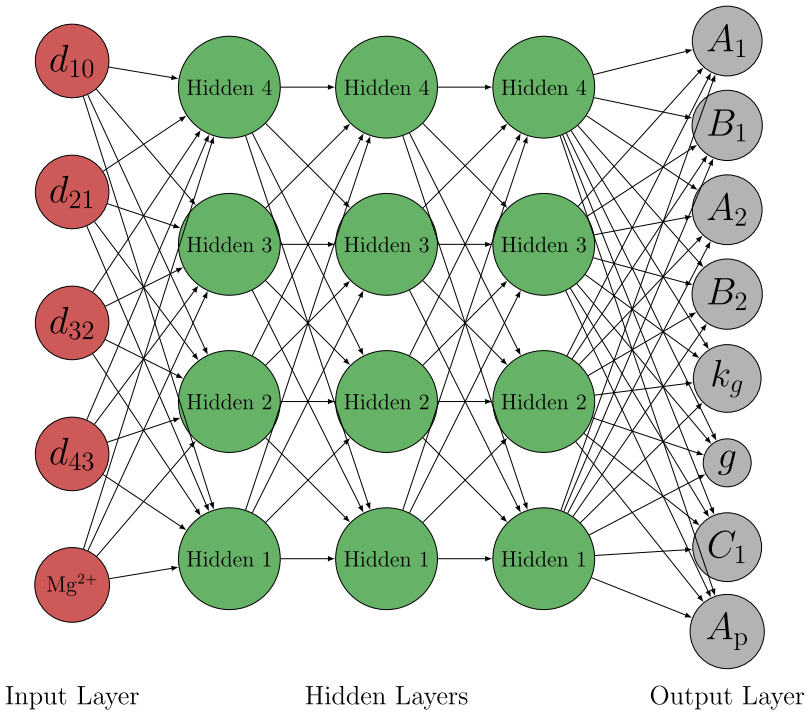


Fig. 5. Schematic representation of the mirror model focusing on inputs and outputs.

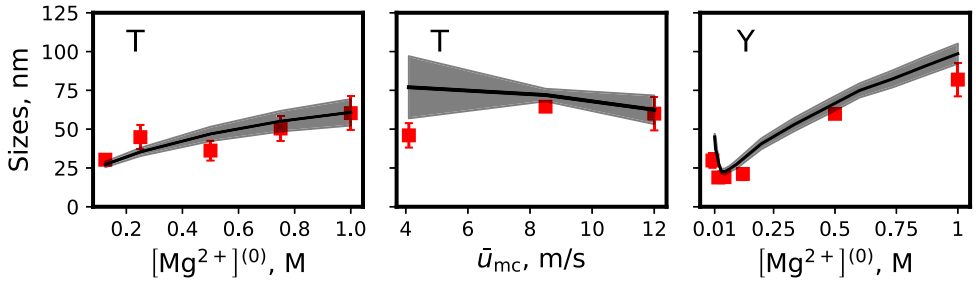
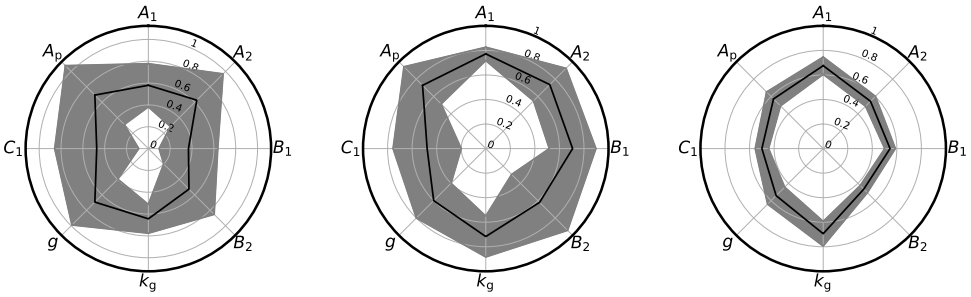


Fig. 6. PBM outcomes vs experimental datasets. Dataset #1 (*left*) is used for fitting, datasets #2 (*center*) and #3 (*right*) are used for validation.

originally part of the fitting stage. Moreover, this new methodology enables a quantitative assessment of the deviation in particle size, represented as a confidence interval. As the concentration increases, the confidence interval expands for both datasets #1 and #3. The rising concentration amplifies the significance of turbulence-related contributions (e.g., turbulent aggregation), making the sizes more prone to variations [33]. Conversely, for dataset #2, the confidence interval widens as the flow rate decreases. When velocity decreases, the uncertainty associated with the RANS turbulent model increases when transitioning from a fully developed turbulent regime to a regime with reduced turbulence [32,43]. Furthermore, although certain combinations of kinetics parameters have successfully replicated dataset #2 experimental trend, particularly at the lower end of the confidence interval (Fig. 6-*center*), these results should be carefully considered. It is important to acknowledge the potential influence of quenching times on the measurements, which may have led to slight deviations from what is typically reported in the literature. Generally, as the flow rate decreases or, equivalently, as mixing time increases, one can anticipate larger particle sizes [44], which aligns with the upper end of the confidence interval. In the end, it is important to emphasize the potential applications of the mirror model. Since no assumptions were made about the model, the numerical dataset could consist of modeling approaches more complex than the one employed here. For example, instead of decoupling the flow and turbulent fields, by using a simplified PBE, as done in this work, the model could be based on three-dimensional fully-coupled CFD-PBM simulations. As a result, the proposed methodology allows for precise adjustments, if required, for any model, regardless of its complexity. Notably, this flexibility is not available with CG and PSO.

Key performance indicators. This section compares the performance of two conventional optimization methods, CG and PSO, with the mirror model. Although CG and PSO are components within the optimization loop as illustrated in Fig. 4 ('Optimization Method' box), both methods are referred to as if they represent the entire optimization loop. The analysis focuses on two Key Performance Indicators (KPIs): (i) the number of times the PBM is called and (ii) the time required to reach a solution. The mirror model generates a kinetic parameter set for each operating condition, specifically the five concentrations of dataset #1 in the fitting dataset. In contrast, classical optimization methods optimize all operating conditions in the fitting dataset simultaneously as outlined in our previous publications [32,33]. Therefore, during each optimization cycle of CG and PSO, only one set of kinetic parameters is obtained, representing the optimized solution for all operating conditions in the dataset. To conduct a thorough comparative performance analysis, classical optimization methods were executed five times. The resulting five kinetic parameter sets allowed for the calculation of the mean values and their standard deviation for the parameters. Considering the substantial variations in the ranges of each parameter, often



(a) CG (b) PSO (c) NN

Fig. 7. $\overline{\overline{\varphi}}$ and confidence interval.

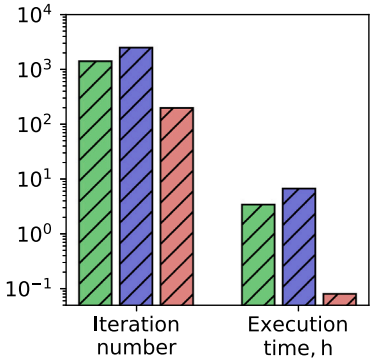


Fig. 8. KPIs for CG (*green*), PSO (*blue*) and NN (*red*).

spanning orders of magnitude, a radar plot (Fig. 7) has been included for visualization. In this plot, each parameter range was normalized between zero and one using the respective minimum and maximum values detailed in Table 2.

The data in Fig. 7-*left* indicates that five optimization cycles are insufficient for CG to generate reliable kinetic parameters with a small standard deviation. This outcome is due to the practice of initializing CG with a completely random initial guess in each cycle. It is worth reiterating that the implementation of this measure was intended to mitigate the influence of user experience on the optimization process and subsequent performance comparison. Additionally, it aimed to ensure that the CG could effectively explore a wide range of the search space. PSO, instead, is acknowledged for its robustness in overcoming the initial guess limitations of CG, consistently yielding more reliable results. As highlighted in Fig. 7-*center*, PSO provides an average kinetic parameter set closer to the one offered by the mirror model (Fig. 7-*right*) and with a smaller standard deviation than the one provided by CG. In addition, variations in PSO results, even after algorithm convergence, can stem from two main factors. Firstly, the problem structure may include multiple optimal solutions or complex topography, leading to locally optimal but not globally optimal solutions. Secondly, the presence of randomness in algorithms like PSO can result in slightly different solutions even post-convergence, due to its stochastic nature. These factors contribute to result variations despite algorithm convergence. However, it proves to be more computationally consuming, particularly when dealing with large optimization problems, as depicted in Fig. 8.

On the left of Fig. 8, the iteration number indicates the number of times the PBM is called. On the right, the cumulative execution time is reported. For CG and PSO, both the iteration number and the execution time represent the total, which is the sum of all five cycles. For the mirror model, the iteration number corresponds to the size of the numerical dataset, which includes 200 simulations (*red* bar on the left). The execution time, instead, covers the dataset generation time, approximately 3 min, and the training time, roughly 5 min (*red* bar on the right). The contrast between the CG-PSO and the mirror model is striking. CG and PSO respectively make roughly 2000 PBM calls (depicted by the *green-blue* bars on the left), a stark contrast to the mirror model's 200 PBM calls. However, the real standout is evident in the execution time. The mirror model accomplishes its tasks in under 10 min, whereas CG and PSO demand nearly 4 h (*green-blue* vs *red* bars on the right). Additionally, it is important to note that CG and PSO would still need more cycles to narrow down the confidence interval, and expert knowledge would remain invaluable in such cases. Classical methods can get trapped in minima that meet the stopping criteria locally but may not be the optimal solution.

6. Conclusions

In this study, we present an innovative deep learning-based approach for multi-variate optimization, with a primary focus on characterizing $Mg(OH)_2$ precipitation kinetics using experimental data. Our research leveraged three distinct datasets: one for PBM fitting and two for PBM validation. The first dataset explored the influence of increasing initial Mg^{2+} concentrations on particle sizes within a T-mixer, while the second dataset delved into the effects of reduced flow rates in the same system. The third dataset examined the impact of increasing initial Mg^{2+} concentrations on particle sizes in a more complex system, featuring a Y-mixer, two divergent channels, and a final pipe. Our methodology exploited a mono-dimensional PBM. This model takes kinetics parameters and concentration as inputs, yielding four characteristic sizes as outputs. We harnessed the PBM to generate a numerical dataset, which was used to train a NN which mirrored the PBM inputs and outputs (consequently called the 'mirror model'). The mirror model took characteristic sizes and Mg^{2+} concentrations as inputs and returned kinetics parameters as outputs. The mirror model played a crucial role in predicting kinetics parameters from experimental sizes from the first dataset. These predictions were used to compute an average set and standard deviation. Surprisingly, the PBM proved highly adept at describing changes in flow rates and substantial reductions in Mg^{2+} concentrations within the new Y-mixer system, even when these conditions were unprecedented. To further evaluate the mirror model's performance, we assessed two KPIs: (i) the iteration number and (ii) the execution time. These KPIs were then compared with those provided by two widely used algorithms, CG and PSO. The outcome of this comparison underscores the enormous potential of the mirror model, with KPIs consistently at least an order of magnitude lower than those of CG and PSO.

GitHub

Population Balance Model
Mirror Model

CRediT authorship contribution statement

Antonello Raponi: Writing – original draft, Writing – review & editing, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Daniele Marchisio:** Writing – review & editing, Methodology, Supervision, Resources, Project administration, Funding acquisition.

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.

Data availability

Data will be made available on request.

Acknowledgments

This study has received funding from the European Union's Horizon 2020 research and innovation programme under Grant Agreement No. 869467 (SEArcularMINE). Financial support came also from ICSC – Centro Nazionale di Ricerca in High Performance Computing, Big Data and Quantum Computing (funded by the European Union – NextGenerationEU). This study was carried out within the «Non-equilibrium self-assembly of structured fluids: a multi-scale engineering problem» project (Project No. 2022JJRH8H) – funded by European Union – Next Generation EU within the PRIN 2022 program (D.D. 104 - 02/02/2022 Ministero dell'Università e della Ricerca). Computational resources were provided by HPC@POLITO, a project of Academic Computing within the Department of Control and Computer Engineering at the Politecnico di Torino (http://www.hpc.polito.it). The authors are also grateful to Prof. Gianluca Boccardo for the fruitful discussions.

This output reflects only the author's view. The European Health and Digital Executive Agency (HaDEA) and the European Commission cannot be held responsible for any use that may be made of the information contained therein.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.cej.2024.151149. This material includes details on the neural network training, influence of parameter selection randomness and PBM.

References

[1] T.E. Graedel, R. Barr, C. Chandler, T. Chase, J. Choi, L. Christoffersen, E. Friedlander, C. Henly, C. Jun, N.T. Nassar, D. Schechner, S. Warren, M. yu Yang, C. Zhu, Methodology of metal criticality determination, Environ. Sci. Technol. 46 (2012) 1063–1070.

[2] T.E. Graedel, E.M. Harper, N.T. Nassar, P. Nuss, B.K. Reck, Criticality of metals and metalloids, Proc. Natl. Acad. Sci. 112 (2015) 4257–4262.

[3] H. Béarat, M.J. McKelvy, A.V. Chizmeshya, R. Sharma, R.W. Carpenter, Magnesium hydroxide dehydroxylation/carbonation reaction processes: Implications for carbon dioxide mineral sequestration, J. Am. Ceram. Soc. 85 (2002) 742–748.

[4] X. Chen, J. Yu, S. Guo, Structure and properties of polypropylene composites filled with magnesium hydroxide, J. Appl. Polym. Sci. 102 (2006) 4943–4951.

[5] H. Gui, X. Zhang, W. Dong, Q. Wang, J. Gao, Z. Song, J. Lai, Y. Liu, F. Huang, J. Qiao, Flame retardant synergism of rubber and Mg(OH)2 in EVA composites, Polymer 48 (2007) 2537–2541.

[6] S. Zhang, F. Cheng, Z. Tao, F. Gao, J. Chen, Removal of nickel ions from wastewater by Mg(OH)2/MgO nanostructures embedded in Al2O3 membranes, J. Alloys Compd. 426 (2006) 281–285.

[7] H. Cao, H. Zheng, J. Yin, Y. Lu, S. Wu, X. Wu, B. Li, Mg(OH)2 complex nanostructures with superhydrophobicity and flame retardant effects, J. Phys. Chem. C 114 (2010) 17362–17368.

[8] C.M. Tai, R.K. Li, Studies on the impact fracture behaviour of flame retardant polymeric material, Mater. Des. 22 (2001) 15–19.

[9] S. Kakaraniya, C. Kari, R. Verma, A. Mehra, Gas absorption in slurries of fine particles: So2 - Mg(OH) 2 - MgSO3 system, Ind. Eng. Chem. Res. 46 (2007) 1904–1913.

[10] A. Sierra-Fernandez, L.S. Gomez-Villalba, O. Milosevic, R. Fort, M.E. Rabanal, Synthesis and morpho-structural characterization of nanostructured magnesium hydroxide obtained by a hydrothermal method, Ceram. Int. 40 (8 PART A) (2014) 12285–12292.

[11] X. Song, K. Tong, S. Sun, Z. Sun, J. Yu, Preparation and crystallization kinetics of micron-sized mg(OH)2 in a mixed suspension mixed product removal crystallizer, Frontiers of Chemical Science and Engineering 7 (2013) 130–138.

[12] A. Cipollina, M. Bevacqua, P. Dolcimascolo, A. Tamburini, A. Brucato, H. Glade, L. Buether, G. Micale, Reactive crystallisation process for magnesium recovery from concentrated brines, Desalin. Water Treat. 55 (2015) 2377–2388.

[13] M. Shiea, A. Querio, A. Buffo, G. Boccardo, D. Marchisio, CFD-PBE modelling of continuous Ni-Mn-Co hydroxide co-precipitation for Li-ion batteries, Chem. Eng. Res. Des. 177 (2022) 461–472.

[14] T. Schikarski, M. Avila, H. Trzenschiok, A. Güldenpfennig, W. Peukert, Quantitative modeling of precipitation processes, Chem. Eng. J. 444 (2022) 136195.

[15] S. Romano, S. Trespi, R. Achermann, G. Battaglia, A. Raponi, D. Marchisio, M. Mazzotti, G. Micale, A. Cipollina, The role of operating conditions in the precipitation of magnesium hydroxide hexagonal platelets using NaOH solutions, Cryst. Growth Des. 23 (2023) 6491–6505.

[16] L. Lu, Q. Hua, J. Tang, Y. Liu, L. Liu, B. Wang, Reactive crystallization kinetics of magnesium hydroxide in the Mg(NO 3) 2 -NaOH system, Cryst. Res. Technol. 53 (2018) 1700130.

[17] A. Guisasola, J. Baeza, J. Carrera, G. Sin, P. Vanrolleghem, J. Lafuente, The influence of experimental data quality and quantity on parameter estimation accuracy, Educ. Chem. Eng. 1 (2006) 139–145.

[18] W. Li, I. Demir, D. Cao, D. Jöst, F. Ringbeck, M. Junker, D.U. Sauer, Data-driven systematic parameter identification of an electrochemical model for lithium-ion batteries with artificial intelligence, Energy Storage Mater. 44 (2022) 557–570.

[19] M. Rezaie, K. karamnejadi azar, A. kardan sani, E. Akbari, N. Ghadimi, N. Razmjooy, M. Ghadamyari, Model parameters estimation of the proton exchange membrane fuel cell by a Modified Golden Jackal Optimization, Sustain. Energy Technol. Assess. 53 (2022) 102657.

[20] Y. Zhang, Y. Tian, X. Zhang, Improved sparseea for sparse large-scale multi-objective optimization problems, Complex Intell. Syst. 9 (2023) 1127–1142.

[21] G. Ma, H. Lin, W. Jin, D. Han, Two modified conjugate gradient methods for unconstrained optimization with applications in image restoration problems, J. Appl. Math. Comput. 68 (2022) 4733–4758.

[22] A.H. Ibrahim, P. Kumam, A. Kamandi, A.B. Abubakar, An efficient hybrid conjugate gradient method for unconstrained optimization, Optim. Methods Softw. 37 (2022) 1370–1383.

[23] M. Schwaab, E.C. Biscaia Jr., J.L. Monteiro, J.C. Pinto, Nonlinear parameter estimation through particle swarm optimization, Chem. Eng. Sci. 63 (2008) 1542–1552.

[24] A.G. Gad, Correction to: Particle swarm optimization algorithm and its applications: A systematic review, Arch. Comput. Methods Eng. 30 (2023) 3471.

[25] Y. Chen, Z. Ye, B. Gao, Y. Wu, X. Yan, X. Liao, A robust adaptive hierarchical learning crow search algorithm for feature selection, Electronics 12 (2023) 3123.

[26] V. Kumar, S.M. Yadav, A state-of-the-art review of heuristic and metaheuristic optimization techniques for the management of water resources, Water Supply 22 (2022) 3702–3728.

[27] B.A. Attea, A.D. Abbood, A.A. Hasan, C. Pizzuti, M. Al-Ani, S. Özdemir, R.D. Al-Dabbagh, A review of heuristics and metaheuristics for community detection in complex networks: Current usage, emerging development and future directions, Swarm Evol. Comput. 63 (2021) 100885.

[28] J. Kennedy, R. Eberhart, Particle swarm optimization, in: Proceedings of ICNN'95-International Conference on Neural Networks, Vol. 4, IEEE, 1995, pp. 1942–1948.

[29] F. Duan, F. Song, S. Chen, M. Khayatnezhad, N. Ghadimi, Model parameters identification of the PEMFCs using an improved design of crow search algorithm, Int. J. Hydrogen Energy 47 (2022) 33839–33849.

[30] I. Pan, L.R. Mason, O.K. Matar, Data-centric engineering: integrating simulation, machine learning and statistics. Challenges and opportunities, Chem. Eng. Sci. 249 (2022) 117271.

[31] A.I.J. Forrester, A. Sóbester, A.J. Keane, Engineering Design via Surrogate Modelling, Wiley, 2008.

[32] A. Raponi, S. Romano, G. Battaglia, A. Buffo, M. Vanni, A. Cipollina, D. Marchisio, Computational modeling of magnesium hydroxide precipitation and kinetics parameters identification, Cryst. Growth Des. 23 (7) (2023) 4748–4759.

[33] A. Raponi, R. Achermann, S. Romano, S. Trespi, M. Mazzotti, A. Cipollina, A. Buffo, M. Vanni, D. Marchisio, Population balance modelling of magnesium hydroxide precipitation: Full validation on different reactor configurations, Chem. Eng. J. (2023) 146540.

[34] D.L. Marchisio, J.T. Pikturna, R.O. Fox, R.D. Vigil, A.A. Barresi, Quadrature method of moments for population-balance equations, AIChE J. 49 (2003) 1266–1276.

[35] D.L. Marchisio, R.D. Vigil, R.O. Fox, Quadrature method of moments for aggregation-breakage processes, J. Colloid Interface Sci. 258 (2003) 322–334.
[36] T. Schikarski, H. Trzenschiok, W. Peukert, M. Avila, Inflow boundary conditions determine T-mixer efficiency, React. Chem. Eng. 4 (2019) 559–568.
[37] R. David, P. Marchal, J.P. Klein, J. Villermaux, Crystallization and precipitation engineering-III. A discrete formulation of the agglomeration rate of crystals in a crystallization process, Chem. Eng. Sci. 46 (1991) 205–213.
[38] A. Mersmann, Crystallization Technology Handbook, Marcel Dekker, 2001, p. 832.
[39] H.Y. Wang, J.D. Ward, Seeding and optimization of batch reactive crystallization, Ind. Eng. Chem. Res. 54 (2015) 9360–9368.
[40] P.H. Karpiński, J. Bałdyga, Precipitation processes, Cambridge University Press, 2019, pp. 216–265,
[41] S. Bhandarkar, R. Brown, J. Estrin, Studies in rapid precipitation of hydroxides of calcium and magnesium, J. Cryst. Growth 97 (1989) 406–414.
[42] P.M. Orlewski, M. Mazzotti, Modeling of mixing-precipitation processes: Agglomeration, Chem. Eng. Technol. 43 (6) (2020) 1029–1039.
[43] C. Walker, A. Manera, B. Niceno, M. Simiano, H.M. Prasser, Steady-state RANS-simulations of the mixing in a T-junction, Nucl. Eng. Des. 240 (2010) 2107–2115.
[44] G. Battaglia, S. Romano, A. Raponi, D. Marchisio, M. Ciofalo, A. Tamburini, A. Cipollina, G. Micale, Analysis of particles size distributions in Mg(OH)2 precipitation from highly concentrated MgCl2 solutions, Powder Technol. 398 (2022) 117106.