Real-time prediction of aggregate structures based on bivariate aerosol dynamics

Lukas Fuchs[1,*,†], Jonah V. Weidemann[2,†], Ivan Skenderović[2], Danijel Čuturić[3], Orkun Furat[1,4], Steven X. Ding[3], F. Einar Kruis[2], Volker Schmidt[1]

[1] Institute of Stochastics, Ulm University, Helmholtzstr. 18, 89069 Ulm, Germany

[2] Institute of Technology for Nanostructures (NST), Center for Nanointegration Duisburg-Essen (CENIDE), University Duisburg-Essen, Bismarckstr. 81, 47057 Duisburg, Germany

[3] Institute for Automatic Control and Complex Systems (AKS), University Duisburg-Essen, Bismarckstr. 81, 47057 Duisburg, Germany

[4] SDU Applied AI and Data Science Unit, University of Southern Denmark, Campusvej 55, 5230 Odense, Denmark

[*] Corresponding author: lukas.fuchs@uni-ulm.de

[†] These authors contributed equally to this work

Abstract

Precise control over nanoparticle synthesis in gas-phase processes such as flame and plasma reactors remains a significant challenge because of the complex, non-linear particle formation dynamics governed by coagulation and sintering. This paper presents a computational methodology that combines a Monte Carlo (MC) simulation framework and a convolutional neural network (CNN)-based surrogate model to accelerate predictions of bivariate particle descriptor vector distributions. The MC framework, optimized for computational efficiency, predicts the evolution of the particle surface area and volume distributions over time under isothermal conditions. This bivariate description enables accurate representation of particle morphology, which in turn influences formation dynamics and final product performance. Evaluation against established models demonstrates high agreement, emphasizing its precision in capturing particle formation dynamics. Indications and restrictions are identified for the achievement of a self-preserving size distribution (SPSD) for both aggregate volume and surface area, offering the potential to simplify and facilitate bivariate modeling approaches. The CNN-based surrogate model leverages bivariate histograms to predict time-dependent distributions for variable temperatures, achieving a 15 000-fold reduction in computation time compared to the MC framework and thus reaching real-time capability, while maintaining sufficient accuracy. In addition, the differentiable nature of the model enables the optimization of temperature profiles. This paper demonstrates the potential for integrating advanced MC frameworks with neural networks to balance computational efficiency and predictive accuracy.

Keywords: bivariate probability distribution, time series prediction, convolutional neural network, particle property, aggregate formation, Monte Carlo simulation

1 Introduction

Precise control over process parameters in gas-phase synthesis of nanoparticles is critical, as variations in the distributions of particle sizes and shapes profoundly influence their functionality and potential applications [22]. However, achieving this level of control remains a formidable challenge, particularly in processes that adhere to complex non-linear dynamics, such as spark- or arc-discharge synthesis [55, 54]. One central challenge is the development of adaptable, real-time prediction models for particle formation, which can guide effective control strategies in synthesis reactors.

Various numerical methods have been proposed to predict the dynamics of particle formation in the gas phase [18, 2, 13, 37, 33], each with varying degrees of complexity, computational cost, and precision. Among these, bivariate models [69, 67, 1, 35], which track the evolution of both particle surface area and volume, are particularly well suited for processes dominated by coagulation and sintering. These models enable the derivation of key morphological descriptors, such as the primary particle size and fractal dimension, while accounting for the effects of irregular shape on formation dynamics. Although the self-preserving size distribution (SPSD) of particle volume is well-established [61] and simplifies predictive modeling, the SPSD of particle surface area has received comparatively less attention. Notable exceptions include studies reported in [56] and [67]. They suggest indications for the existence of a joint SPSD for surface area and volume, which could significantly improve predictions and potentially streamline process design.

Efforts to reduce the computational cost of predictive models have resulted in a range of approaches, each with specific trade-offs. Monodisperse models are computationally efficient but lack accuracy due to the neglect of polydispersity [35]. Method-of-moments models offer greater speed, but rely on simplifications and a priori assumptions on underlying distributions, which can limit their accuracy in capturing complex dynamics [67]. Discrete sectional models, such as that proposed in [69], explicitly resolve the particle size distribution (PSD) by dividing particle sizes into sections and solving population balance equations for each. This yields accurate results but is computationally demanding. Simplifications, such as reducing sectional resolution or dimensionality while maintaining accuracy, have been carried out, for example, in [41] and [57]. In [41], a bivariate 1D sectional model has been successfully coupled with computational fluid dynamics simulations, enhancing its applicability to practical scenarios.

The Monte Carlo (MC) framework offers an alternative approach, using stochastic sampling to realistically mimic the probabilistic nature of particle coagulation. Furthermore, MC frameworks enable seamless multi-property tracking of particles, making them a powerful tool for modeling complex formation dynamics. However, they are traditionally computationally intensive. Recent advances, such as GPU parallelization [66], weighted particles [21], stochastic resolution concepts, and particle merging techniques [32], have significantly improved computational efficiency. However, achieving the level of speed required for real-time applications while maintaining high accuracy remains a challenge. The present study builds on these advances, focusing on the further acceleration of particle formation predictions.

To address the computational challenges associated with MC simulations of particle aggregation, the present paper proposes the use of a surrogate model [64, 40] based on convolutional neural networks (CNNs) [39]. CNNs have proven effective for fast image-based regression, segmentation, and classification [30, 16, 17, 27, 34], whereas surrogate models offer a computationally efficient alternative by approximating the output of detailed simulations with significantly reduced computational effort. Thus, the primary motivation for developing a CNN-based surrogate model is to accelerate the simulation process while maintaining high precision in predicting the outcomes of particle aggregation.

Building on this, CNNs also enable real-time adaptability by providing high-speed predictions that facilitate real-time feedback-driven adjustments to synthesis conditions. These dynamic adjustments are particularly relevant for process control in arc-discharge synthesis, where maintaining stability under fluctuating plasma conditions is crucial. Previous applications of CNNs in modeling time-dependent systems, such as gas spreading [43] and differential equation solving [44], highlight their versatility and effectiveness. Furthermore, recent advances in data-driven process monitoring, such as deep neural network-aided canonical correlation analysis (DNN-CCA) [9] and canonical correlation deep neural network (CCDNN) [10], have demonstrated their potential in capturing multivariate dependencies in dynamic systems, a critical feature for identifying and mitigating disruptions in real-time. The primary strength of both methods lies in correlation analysis, monitoring and fault detection, improving feature extraction and multivariate data handling, rather than predicting high-speed time series.

In particular, so-called autoencoders have been used in process optimization. An autoencoder [58] is a neural network that learns to encode input data into a lower-dimensional representation and then decodes it back to its original form, capturing essential features in the process. Advanced approaches, such as variational discriminative stacked autoencoders, have demonstrated enhanced feature representation capabilities by incorporating a pre-learned discriminator, improving fault detection and process optimization in industrial settings [26]. Autoencoder methods, widely explored in industrial process monitoring, provide a solid foundation to capture essential dynamics while enabling robust fault detection and parameter optimization [38].

The surrogate model developed in the present paper aims to predict two-dimensional distributions of surface area and volume of aggregated particles over time, conditioned on temperature. This allows for the prediction of particle properties over time and temperature optimization to achieve desired aggregate properties. More precisely, we propose a framework to predict the evolution of the distribution of particle surface area and volume one time step ahead. For this, an autoencoder is trained to predict a series of two-dimensional histograms of particle sphericity, a particle descriptor used to quantify the relative aggregate surface area [63], and aggregate volume for a given reactor temperature. This approach is implemented for the proposed small autoencoder network architecture (SAN) as well as for two network architectures from the literature which were designed to perform similar tasks but in very different contexts. The first is an Atari frame prediction network (AFPN) which was designed to predict a series of frames of Atari games, conditioned on player input [45]. The second network was designed to predict human movement actions in videos, conditioned on the elapsed time [62], which will be referred to as video frame prediction network (VFPN).

The purpose of the present paper is to answer the following research questions. Are the bivariate formation dynamics accurately described by the present MC framework, and under which conditions is the attainment of a SPSD for both volume and surface area observed? Is it possible to accurately predict temperature-dependent particle properties over time in gas-phase synthesis of nanoparticle aggregates? Can this prediction be performed at a speed that is feasible for application in autonomous process control? That is, can even a low-parametric neural network architecture achieve high predictive accuracy in this task?

The remainder of this paper is organized as follows. First, in Section 2.1, the particle formation model based on coagulation and sintering is introduced. Afterwards, the MC framework that is used as a basis for neural network training is summarized in Section 2.2. Next, in Section 2.3, the surrogate model and, in particular, the acquisition of training data, the data transformation, the network architecture, and the training procedure are described. The MC framework is then evaluated in Section 3.1, followed by an investigation of distribution dynamics in Section 3.2. In addition, in Section 3.3, the predicted time series of particle descriptor vector distributions for different surrogate model architectures and losses are compared, and their applicability in process control is discussed. Finally, Section 4 concludes.

2 Methods

2.1 Particle formation dynamics

In gas-phase particle formation, colliding particles adhere upon contact, in a process known as coagulation. Thermal energy then causes restructuring of material between the individual constituent primary particles in a process referred to as sintering. This results in a reduction of the surface area of the aggregates, which has a direct influence on their collision radius [35]. The collision radius determines the probability of collisions, influencing the overall growth dynamics [28]. Moreover, the collision radius is influenced by coagulation due to the addition of colliding particle volumes and surface areas. The interconnected nature of volume and surface area evolution thus necessitates their simultaneous prediction in synthesis processes with significant thermal energy input, such as flame spray pyrolysis, hot-wall reactors, spark-discharge, or arc-discharge.

The present study does not explicitly model nucleation, as it assumes that the smallest particles formed are thermodynamically stable in the absence of a nucleation barrier. This assumption is supported by studies on gas-phase reactors for materials such as SiO_2 [59] and TiO_2 [68]. In these reactor types, the high degree of supersaturation results in a small critical nucleus size, as noted by Schmidt-Ott for spark-discharge reactors [50]. In aerosols with high volume fractions, particle growth is predominantly governed by ballistic and diffusional coagulation, rendering other mechanisms such as nucleation and condensation negligible [7]. For modeling purposes, initial primary particles are assumed to be instantaneously introduced into the system and homogeneously distributed. This simplifies the modeling process by decoupling monomer release dynamics from particle growth mechanisms. The following sections provide a detailed description of the coagulation and sintering models used in this paper.

2.1.1 Coagulation

Let $N(v,t)$ denote the (expected) number of particles per unit reactor volume (number concentration) at time $t \geq 0$ such that the volume of individual particles does not exceed the threshold $v > 0$. Assume that $N(v,t)$ can be written as integral, i.e., for some function $n\colon (0,\infty) \times [0,\infty) \to [0,\infty)$ we have

$$N(v,t) = \int_0^v n(v',t) \mathrm{d}v', \tag{1}$$

for any $v, t > 0$, where $n(v',t)\mathrm{d}v'$ represents the (differential) number concentration of particles with volume between v' and $v' + \mathrm{d}v'$ at time t. The net rate of change of the values of $n(v,t)$ with respect to time t is governed by a population balance equation (PBE), which is modeled by the classical Smoluchowski expression [53] for the coagulation of spherical particles. Under the assumption of binary collisions and spatial homogeneity, it is given as

$$\frac{\mathrm{d}n(v,t)}{\mathrm{d}t} = \frac{1}{2} \int_0^v \beta\left(v - v', v'\right) n\left(v - v', t\right) n\left(v', t\right)\, \mathrm{d}v' - n(v,t) \int_0^\infty \beta\left(v, v'\right) n(v',t)\, \mathrm{d}v', \tag{2}$$

for any $v, t > 0$, and a particle collision rate $\beta\colon (0,\infty)^2 \to [0,\infty)$. Note that in deterministic approaches such as discrete-sectional schemes [37], the PBE given in Eq. (2) is commonly discretized in particle volume. In contrast, the stochastic nature of the MC framework considered in the present work bypasses this by implicitly bounding the domain through sampling, thereby eliminating the need for an explicit discretization grid of the particle volume [42]. The first term on the right-hand side of Eq. (2) accounts for all possible combinations of collisions between particles with volumes v' and $v - v'$ that result in the formation of particles with volume v, while the second term represents the loss of particles with volume v due to their collision with other particles. The dynamics of this process are strongly influenced by the particle collision rate, β, commonly referred to as the coagulation kernel $\beta(v_i, v_j)$ for two colliding particles with volumes v_i and v_j, respectively.

In high-temperature systems, the free-molecular regime dominates. In this regime, β is defined as the product of the relative velocity of the colliding particles and their combined geometric cross-sectional area. For spherical particles, the cross-sectional area is the smallest, whereas for fractal-like agglomerates, such as those formed by ballistic cluster-cluster agglomeration (BCCA), the cross-sectional area increases significantly. These agglomerates are characterized by a fractal dimension D_f of approximately 1.91 [12]. The fraction of the surface area a of a particle, which is exposed to the surrounding and therefore accessible for collisions with other particles, is called surface area accessibility factor $s \in [\frac{2}{3}, 1]$ and needs to be included in the computation of collision cross-section. In [69], a general formulation of the coagulation kernel has been provided for two colliding particles of any shape, which is utilized in the present work, ensuring accurate modeling of collision dynamics in diverse particle systems. This is done by incorporating the additional dependence of β on both a and s. The domain of the coagulation kernel β is therefore extended to $\beta\colon (0,\infty)^2 \times (0,\infty)^2 \times [\frac{2}{3}, 1]^2 \to [0,\infty)$, where

$$\beta\left(v_i, v_j, a_i, a_j, s_i, s_j\right) = \frac{1}{4}\left(\frac{8k_\mathrm{B}T}{\pi\rho_\mathrm{p}v_i} + \frac{8k_\mathrm{B}T}{\pi\rho_\mathrm{p}v_j}\right)^{\frac{1}{2}}\left(\sqrt{s_i a_i} + \sqrt{s_j a_j}\right)^2, \tag{3}$$

for any $v_i, v_j, a_i, a_j > 0$ and $s_i, s_j \in [\frac{2}{3}, 1]$. Here, $k_\mathrm{B} > 0$ denotes the Boltzmann constant, $T > 0$ is the temperature, and $\rho_\mathrm{p} > 0$ represents the particle mass density, which corresponds to the bulk density of the material.

Furthermore, the fractal nature of particles is often incorporated into β by predefining a constant D_f, which scales the collision radius according to a power-law relationship [28]. However, this approach assumes a fixed value of D_f, independent of the size of the agglomerates or its distribution, which can lead to inaccuracies in modeling complex systems. Following the approach proposed in [69], we introduce a surface fractal dimension D_s initialized at 2.0. This quantity can be dynamically computed for each particle based on the ratio of volume to surface area, normalized by the volume v_0 and surface area a_0 of the initial primary particles, i.e., $D_\mathrm{s}(v, a)$ is obtained as a solution of

$$\frac{a}{a_0} = \left(\frac{v}{v_0}\right)^{\dfrac{D_\mathrm{s}}{3}}. \tag{4}$$

However, note that this relationship is only used for large agglomerates with a number of primary particles $n_\mathrm{p} > 10$ [69]. Smaller agglomerates are assumed to be approximately spherical ($D_\mathrm{s} = 2.0$), therefore their full surface area is accessible for collisions, i.e. $s = 1$. The lowest accessibility of $s = 2/3$ is obtained for close-packed spheres [49], for large n_p and a maximum surface fractal dimension of $D_\mathrm{s} = 3.0$. A linear interpolation between the boundary limits of $D_\mathrm{s}(v, a)$ as suggested in [69] gives a relationship between s and the agglomerate size, represented by n_p, yielding

$$s = \begin{cases} 1, & \text{if } n_{\mathrm{p}} < 10, \\ (D_{\mathrm{s}} - 2)\left(\dfrac{2}{n_{\mathrm{p}}}\right)^{1-\alpha} + 3 - D_{\mathrm{s}}, & \text{if } 10 \leq n_{\mathrm{p}} \leq 317, \\ \dfrac{2}{3}, & \text{if } n_{\mathrm{p}} > 317, \end{cases} \tag{5}$$

where the surface area scaling factor α is set to 0.92, consistent with the findings given in [49] for silver agglomerates. While some studies suggest that α may also depend on agglomerate size [3], this effect primarily influences small agglomerates with $n_{\mathrm{p}} < 10$. For these cases, we assume full surface area accessibility, i.e. $s = 1$, as mentioned previously. The dynamic variation of s by Eq. (5) directly modifies the coagulation kernel in Eq. (3), thereby influencing the collision rates and subsequently the temporal evolution of particle concentrations described by the PBE given in Eq. (2).

Finally, we consider the approximate characteristic coagulation time τ_{c} that describes the average time between collisions. Note that the characteristic coagulation time is often approximated by $\tau_{\mathrm{c,mono}}$ assuming a monodisperse aerosol, where $\tau_{\mathrm{c,mono}}$ is given as the inverse of the initial (total) concentration $N_0 = \lim_{v\to\infty} N(v,0)$ multiplied by the uniform, monodisperse collision rate β_{mono} [4], i.e.,

$$\tau_{\mathrm{c,mono}} = \frac{2}{N_0 \beta_{\mathrm{mono}}}. \tag{6}$$

This approximation can be refined for a polydisperse aerosol by estimating the average effective collision rate $\bar{\beta} \in (0, \infty)$, instead of β_{mono}, over a short time interval $[t_i, t_{i+1}]$ by considering the change of the total particle concentration from N_{t_i} to $N_{t_{i+1}}$ between times t_i and t_{i+1}, respectively [23], so that $\bar{\beta}$ is computed as

$$\bar{\beta} = \frac{2\left(\frac{1}{N_{t_{i+1}}} - \frac{1}{N_{t_i}}\right)}{t_{i+1} - t_i}. \tag{7}$$

Assuming that τ_{c} remains constant over the time period from t_i to t_{i+1}, which is acceptable for small intervals, from Eq. (6) and putting $N_0 = N_{t_i}$ we obtain

$$\tau_{\mathrm{c}} = \frac{N_{t_{i+1}}\left(t_{i+1} - t_i\right)}{N_{t_i} - N_{t_{i+1}}}. \tag{8}$$

2.1.2 Sintering

The modeling approach presented in Section 2.1.1 allows the coagulation dynamics to adjust to the structural changes experienced throughout the simulation, which makes it suitable and accurate to predict real-world system applications. In particular, this allows us to consider the following effect.

At high temperatures, particle formation is accompanied by material restructuring via thermal energy-driven atomic diffusion, in a process called sintering, which reduces the aggregate surface area. In [31], it has been suggested that the rate of changes in the surface area of the particles should be represented as an exponential relaxation process driven by the thermodynamic tendency to minimize surface energy. Thus, if no coagulation occurs, the change in the aggregate surface area $a = a(t)$ over time t is given by

$$\frac{\mathrm{d}a(t)}{\mathrm{d}t} = -\frac{1}{\tau_{\mathrm{s}}(t)}\left(a(t) - a_{\mathrm{sph}}(t)\right), \tag{9}$$

where $a_{\mathrm{sph}} = a_{\mathrm{sph}}(t)$ is the surface area of a sphere with the volume $v = v(t)$ of the aggregate at time t, and $\tau_{\mathrm{s}} = \tau_{\mathrm{s}}(t) > 0$ is the characteristic sintering time (at time t), which is highly sensitive to temperature. It is defined as the time required to reduce the surface area of an aggregate to approximately 63%, reaching the equilibrium surface area a_{sph} of a perfect sphere.

Note that from Eq. (4) with $D_{\mathrm{s}} = 2.0$, we get

$$a_{\mathrm{sph}} = a_0 \left(\frac{v}{v_0}\right)^{\frac{2}{3}}. \tag{10}$$

The characteristic sintering time τ_{s} is often given as an effective sintering time by fitting experimental data from well-defined setups or molecular dynamics simulation results to an Arrhenius equation, i.e.,

$$\tau_{\mathrm{s}} = A_s T^b d_{\mathrm{p}}^m \exp\left(\frac{E_a}{RT}\right), \tag{11}$$

for some exponents $b \in \{0, 1\}$ and $m > 0$, where the prefactor $A_s > 0$ and the activation energy $E_a > 0$ are material constants, $R > 0$ is the universal gas constant and T the temperature. Furthermore, $d_{\mathrm{p}} > 0$ is the effective diameter of the (overlapping) monodisperse primary particles, an aggregate is considered to be represented by.

Thus, by means of Eq. (11), the primary sintering mechanism provides a relationship between τ_{s} and d_{p} via the exponent m. The present study considers grain boundary diffusion sintering of both Si [35] and Fe [47] that leads to $m = 4$.

The reduction of the surface area is assumed to homogeneously restructure the particle-forming aggregates through sinter neck formation. In contrast to pure coagulation, where the primary particle diameter does not change and remains equal to their initial values (v_0 and a_0), sintering leads to a reduction in aggregate surface area and thus to an increase in volume-to-surface area ratio. To incorporate this effect into the coagulation model while maintaining the assumption that an aggregate is composed of monodisperse (spherical) primary particles [57], the increase in volume-to-surface ratio is represented by a uniform increase in the diameter of the primary particles. As a result, an aggregate is considered to be represented by overlapping monodisperse primary particles with an effective diameter $d_{\mathrm{p}} > 0$. For an aggregate with volume v and surface area a, the diameter d_{p} is defined as the surface area-equivalent primary particle diameter, with

$$d_{\mathrm{p}} = \frac{6v}{a}. \tag{12}$$

The number of primary particles n_{p} is computed for each aggregate as the ratio of v and its primary particle volume v_{p}, corresponding to d_{p}, i.e.,

$$n_{\mathrm{p}} = \frac{v}{v_{\mathrm{p}}} = \frac{6v}{\pi d_{\mathrm{p}}^3}. \tag{13}$$

The aggregate diameter is often given as the diameter d_{v} of the volume-equivalent sphere, or the diameter d_{a} of the surface area-equivalent sphere, where

$$d_{\mathrm{v}} = \left(\frac{6v}{\pi}\right)^{1/3} \qquad \text{and} \qquad d_{\mathrm{a}} = \left(\frac{a}{\pi}\right)^{1/2}. \tag{14}$$

Controlling the ratio

$$\chi = \frac{\tau_{\mathrm{c}}}{\tau_{\mathrm{s}}} \tag{15}$$

of τ_c and τ_s during synthesis is crucial in designing the mesostructure of the particles and therefore influences their desired properties in application [20]. Note that $\chi \to \infty$ leads to perfect spheres by fast sintering (coalescence) while $\chi \to 0$ yields agglomerates of lightly bonded primary particles. Usually, aggregates are formed for a value of χ between 0.1 and 10 [7]. Error propagation of χ is considered in the supplementary material, see B.

2.2 Monte Carlo framework

In the event-driven MC framework used in this paper, the particle population is represented by a set of $N_{\mathrm{SP}} > 0$ so-called simulation particles (SPs), each given by a tuple of descriptors v_i and a_i which are associated with a weight w_i that reflects its concentration, where $i \in \{1, \ldots, N_{\mathrm{SP}}\}$. At the beginning of the simulation, each of the SPs is initialized with equal w and descriptors v_0, a_0, where the total number of simulation particles N_{SP} remains constant throughout the simulation. The framework utilizes an operator splitting approach, in which a coagulation event is executed first, followed by a reduction of the surface area of all particles due to sintering within a given MC time step. This approach decouples the collision and sintering rates, introducing a small error that becomes negligible at sufficiently small time scales [18]. A schematic description of the MC framework is shown in Figure 1.

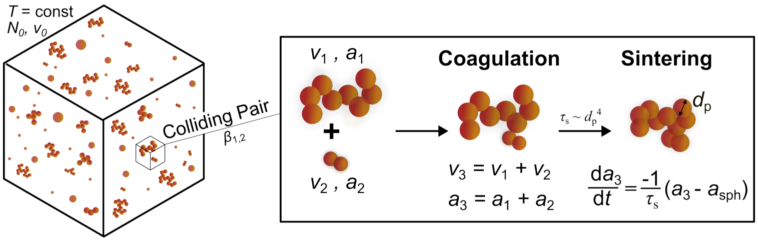


Figure 1: Schematic description of bivariate MC simulation.

The univariate, weighted particle MC framework, which predicts the evolution of the particle volume distribution, is presented in detail in [32], where the key characteristics are summarized (excluding nucleation). The bivariate generalization to multiple descriptors (v, a) is explained later in this section. Note that MC frameworks are well suited for multivariate simulations because of the simplicity in assigning multiple unique properties to each SP (here v and a), allowing a more accurate representation of particle dynamics [32] than by methods by which only one particle descriptor is considered.

Simulations are accelerated using the concept of stochastic resolution described in [32], resulting in optimal allocation of the SP weights $w_1, \ldots, w_{N_{\mathrm{SP}}}$ after coagulation events while ensuring accurate representation. It also yields a symmetric coagulation kernel, which is desirable for numerical consistency. The acceptance-rejection technique of [66] is used to determine the MC time step. In each coagulation step, a representative sample $S \subset \{1, \ldots, N_{\mathrm{SP}}\}^2$ of pairs of SP indices (consisting of N_{Sample} index pairs) is selected at random and evaluated in parallel on GPU threads. For each pair $(i, j) \in S$, using Eq. (3) the corresponding value of the coagulation kernel $\beta_{i,j}$ is computed based on the values of $v_i, v_j, a_i, a_j, s_i, s_j$. Then, for all N_{Sample} pairs, the average β_{mean} is obtained as

$$\beta_{\mathrm{mean}} = \frac{1}{N_{\mathrm{Sample}}} \sum_{(i,j)\in S} \beta_{i,j}. \tag{16}$$

To facilitate rejection sampling, a majorant kernel $\beta_{\max} = \gamma \beta_{\mathrm{mean}}$ is computed by multiplying β_{mean} by a constant $\gamma \gg 1$. The latter should be chosen large enough to ensure $\beta_{\max}$ serves as a valid upper bound for all possible $\beta_{i,j}$, but not so large that it unnecessarily increases the rejection rate and, therefore, the computational time. Then pairs of SP indices $(i,j) \in \{1, \dots, N_{\text{SP}}\}^2$ are sampled uniformly at random, and a uniformly distributed number $r_{i,j} \in (0,1)$ is drawn. A chosen pair (i,j) of SP indices is accepted for coagulation if the inequality

$$r_{i,j} \leq \frac{\beta_{i,j}}{\beta_{\max}} \tag{17}$$

is satisfied. Otherwise, another pair is selected at random and tested. Once a valid coagulation pair is accepted, the coagulation event is carried out, and the corresponding MC time step is determined, advancing the simulation. For further details, see [32, 66].

After each coagulation event, the effective diameter d_{p} is computed for each aggregate using Eq. (12), and the surface area of the particles is reduced according to Eq. (9), based on the MC time step previously determined. This modeling sequence introduces several kinds of approximation. First, while real aggregates typically contain primary polydisperse particles, we reduce this complexity using a single surface area-equivalent primary particle diameter d_{p}. Furthermore, d_{p} is computed for the newly formed aggregates despite the potentially large differences of d_{p} in the previously collided aggregates. It is assumed that d_{p} is constant during the sintering step for each aggregate. In Section 3.1.2 the MC simulations are benchmarked against literature methods to assess the inaccuracies arising from the aforementioned assumption. For example, more detailed approaches towards the primary particle size distribution and sintering dynamics in MC simulations are given in [48]. However, this kind of complexity significantly increases computational effort, hindering the collection of a comprehensive data set necessary for adequate training of the surrogate model described in Section 2.3.

The solution of Eq. (9) is found numerically using the differential equation solver CVode from the SUNDIALS suite [24]. For constant reactor temperatures (as observed in the present paper), implementing the analytical solution for Eq. (9) is also perfectly sufficient. More information is given in the supplementary material, see A.

The accuracy of the proposed method depends on the number of SPs used in the simulations. To further improve the results, in addition to increasing the number N_{SP} of SPs, the MC simulation can also be executed multiple times in parallel, where the number of parallel executions is denoted by $N_{\text{Sims}} > 0$. This effectively reduces statistical fluctuations. Using GPU-based parallelization enables the efficient processing of multiple simulations simultaneously with minimal runtime overhead, significantly improving computational efficiency. All simulations are aligned at predefined export points in time (100 in total) to ensure consistency. At each point in time, the particle descriptors are sampled simultaneously across all simulations. These synchronized outputs are then used for data analysis and training of the surrogate model, as discussed in Section 2.3. In the present paper, we put $N_{\text{SP}} = 1\,000$, $N_{\text{Sample}} = 256$, and $\gamma = 1000$.

In order to interpret the output of the MC simulation, which consists of the SPs at a given point in time, we compute a set of well-interpretable scalar metrics. Recall that the MC framework delivers N_{SP} triplets $(v_1, a_1, w_1), \dots, (v_{N_{\text{SP}}}, a_{N_{\text{SP}}}, w_{N_{\text{SP}}}) \in (0,\infty)^3$, representing the volume, surface area, and weight of the SPs. For the volumes $v_1, \dots, v_{N_{\text{SP}}} \in (0,\infty)$, the arithmetic average $\bar{v}$, as well as the geometric mean v_{g} and the geometric standard deviation $\sigma_{\text{g},v}$, are defined as follows:

$$\bar{v} = \left(\sum_{i=1}^{N_{\text{SP}}} w_i \right)^{-1} \sum_{i=1}^{N_{\text{SP}}} w_i v_i, \tag{18}$$

$$v_{\mathrm{g}} = \exp\left[\left(\sum_{i=1}^{N_{\mathrm{SP}}} w_i\right)^{-1} \sum_{i=1}^{N_{\mathrm{SP}}} w_i \ln(v_i)\right], \tag{19}$$

$$\sigma_{\mathrm{g},v} = \exp\left[\sqrt{\left(\sum_{i=1}^{N_{\mathrm{SP}}} w_i\right)^{-1} \sum_{i=1}^{N_{\mathrm{SP}}} w_i \left(\ln(v_i) - \ln(v_{\mathrm{g}})\right)^2}\right]. \tag{20}$$

Analogously, these descriptors are defined for the surface areas $a_1, \ldots, a_{N_{\mathrm{SP}}} \in (0, \infty)$, volume-equivalent diameters $d_{\mathrm{v},1}, \ldots, d_{\mathrm{v},N_{\mathrm{SP}}} \in (0, \infty)$, surface area-equivalent diameters $d_{\mathrm{a},1}, \ldots, d_{\mathrm{a},N_{\mathrm{SP}}} \in (0, \infty)$, primary particle diameters $d_{\mathrm{p},1}, \ldots, d_{\mathrm{p},N_{\mathrm{SP}}} \in (0, \infty)$, and number of primary particles per aggregate $n_{\mathrm{p},1}, \ldots, n_{\mathrm{p},N_{\mathrm{SP}}} \in (0, \infty)$ of the SPs and are denoted by ($\bar{a}$, a_{g}, $\sigma_{\mathrm{g,a}}$), ($\bar{d_{\mathrm{v}}}$, $d_{\mathrm{v,g}}$, $\sigma_{\mathrm{g},d_{\mathrm{v}}}$), ($\bar{d_{\mathrm{a}}}$, $d_{\mathrm{a,g}}$, $\sigma_{\mathrm{g},d_{\mathrm{a}}}$), ($\bar{d_{\mathrm{p}}}$, $d_{\mathrm{p,g}}$, $\sigma_{\mathrm{g},d_{\mathrm{p}}}$), and ($\bar{n_{\mathrm{p}}}$, $n_{\mathrm{p,g}}$, $\sigma_{\mathrm{g},n_{\mathrm{p}}}$) respectively.

All settings and material parameters utilized in MC simulations in this work are summarized in Table 1. The parameter set in row 1 is employed to validate the MC method against literature methods in Sections 3.1.1 and 3.1.2. Self-preserving size distribution (SPSD) dynamics discussed in Section 3.2 are computed using the parameter set in row 2. The dataset used to train the surrogate model (Section 2.3) is generated using the parameter set in row 3.

Table 1: Settings and parameters for the MC simulations performed in the present work, where the parameters for Si are taken of [35], and those for Fe of [47]. Two rows are shown for Fe because the surrogate model (Section 2.3) does not include a dynamic sintering rate and was trained on an extended temperature range compared to the data used in the distribution dynamics analysis in Section 3.2.

Material	ρ_{p}, kg/m^3	v_0, m^3	N_0, 1/m^3	T, K	A_{s}, s/m^4	b	m	E_{a}, J/mol
Si	2330	$3.35 \cdot 10^{-29}$	10^{22}	773, 1073	$1.15 \cdot 10^{13}$	1	4	230000
Fe	7874	$5.24 \cdot 10^{-28}$	$8 \cdot 10^{19}, 8 \cdot 10^{22}$	800 - 1100	$1.12 \cdot 10^{20}$	0	4	185610
Fe	7874	$5.24 \cdot 10^{-28}$	$8 \cdot 10^{19}$	800 - 2120	-	-	-	-

2.3 Surrogate model

The MC framework for predicting the distributions of particle descriptor vectors (v, a) during aggregation, summarized in Section 2.2, provides an interpretable physics-based method; however, it lacks computational efficiency to be applicable in real-time process control. Thus, in this section, we introduce a CNN-based surrogate model that is a black box, but is suitable for real-time applications. The use of neural networks as surrogate models for predicting particle aggregation processes offers significant advantages over the direct application of conventional MC simulations. Primarily, neural networks provide an enormous increase in computational speed, allowing for almost instant predictions once trained, compared to the time-intensive nature of MC simulations. In addition, by predicting distributions of descriptors rather than simulating individual particles, enhanced stability and efficiency are offered, making neural networks a robust tool for modeling complex particle systems.

More precisely, using 2D histograms, the proposed surrogate model predicts the (mass-weighted) bivariate distributions of an aggregate descriptor vector at time steps $t \in \{1, \ldots, 100\}$, given a 2D histogram of the descriptor vector at the preceding time step $t - 1$, and a certain reactor temperature $T > 0$. This means in particular that the proposed surrogate model is able to handle different temperatures without the need for extensive adjustments and, therefore, is suitable for temperature optimization. Thus, the surrogate model can be used to predict temperatures, leading to distributions of the descriptor vector similar to the desired ones. Moreover, due to the differentiability of neural networks, this optimization can be performed efficiently through gradient-descent-based optimization schemes [5].

2.3.1 Acquisition of training data

To develop a data-driven surrogate model, the first step is to collect training data. This is accomplished using the MC framework described in Section 2.2, which predicts the distributions of the aggregate volume v and surface area a for various reactor temperatures $T > 0$ over time. The material parameters used for data acquisition are provided in row 3 of Table 1. For the reactor temperatures T used during data acquisition, it holds that $T \in \{800, 810, \ldots, 2120\}$. As this paper aims to demonstrate a proof-of-concept for CNN-based surrogate models, the sintering dynamics is simplified for now by considering a constant characteristic sintering time of $\tau_{\mathrm{s}} = 10^{-4}s$. Future work will extend the surrogate model to predict the particle formation process using dynamic sintering rates.

The training data for the surrogate model consists of descriptor data generated by the MC simulation framework with a temporal resolution of 0.01 s over a total duration of 1 s. Specifically, the MC framework is used to compute the evolution of the descriptor vector distribution at 100 equidistant points in time for different temperatures T. The collected data is then divided into two subsets, training data (90%) and evaluation data (10%), where each temperature $T \in \{800, 810, \ldots, 2120\}$ appears in only one of these sets.

2.3.2 Data preprocessing

The surrogate model considered in this paper operates on bivariate histograms of descriptor vectors. Since these can be represented as images, they allow for the use of widely studied methods of computer vision [39].

Note that there is a strong correlation between the volume v and the surface area a of the aggregates. As a result, when building a bivariate histogram based on both v and a, only a few bins exhibit heights larger than 0, particularly for small values of v. This sparsity leads to a low precision in these ranges, as observable in Figure 2 a. To address this issue, the descriptor vectors are transformed in a preprocessing step prior to histogram computation. Thus, the sphericity $\Phi \in (0, 1]$ of particles [63] is considered instead of the surface area $a \in (0, \infty)$, where the sphericity Φ of an aggregate with volume v and surface area a is given by

$$\Phi = \frac{\pi^{\frac{1}{3}} (6v)^{\frac{2}{3}}}{a}. \tag{21}$$

This transformation is invertible with respect to a and, using a 2D histogram-based representation of descriptor vectors, offers a more precise depiction of low-volume particles by considering the descriptors v and Φ that are less correlated than v and a, see Figure 2 b. However, one problem remains; i.e., the stochastic nature of the MC framework introduces variability. Running the MC simulation twice and comparing the resulting histograms bin by bin can result in significant errors, particularly for large but rarely occurring aggregates ($\frac{v}{v_0} > 40$) with similar but slightly different sphericities, which would be assigned to different bins.

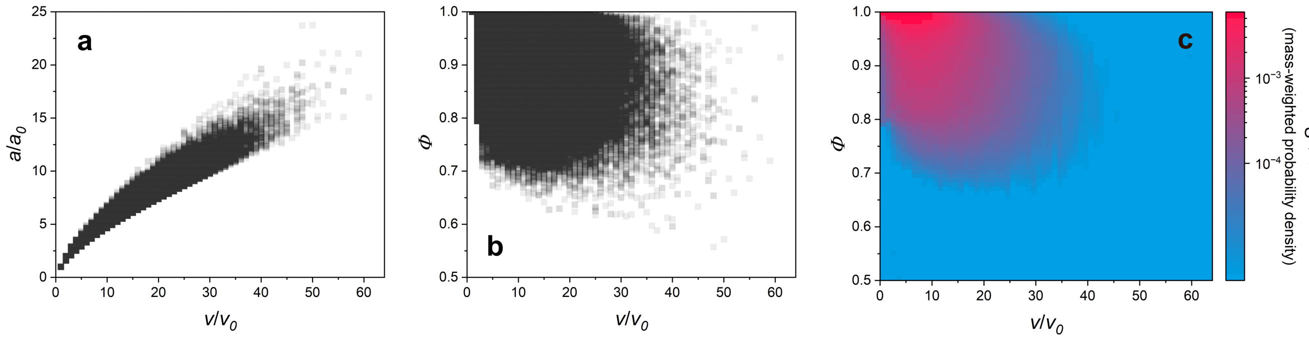


Figure 2: (a) Scatter plot of volume v and surface area a, (b) corresponding scatter plot of volume v and sphericity ϕ, and (c) bivariate histogram $S^{(t,T)}$ corresponding to the scatter plot shown in (b). The values of the histogram entries are given in logarithmic scale for visualization purposes.

To reduce the noise in the simulated histograms, we distribute the probability mass of each aggregate between neighboring bins within the same volume v, rather than assigning it to a single bin. Specifically, let $X^{(t,T)} \subset (0,\infty) \times (0,1]$ be the set of descriptor vectors (v,Φ) obtained from the MC simulation at temperature T and time t. Furthermore, for each $i \in \mathbb{N} = \{1,2,\ldots\}$, let $f_i^{(t,T)} \colon \mathbb{R} \to [0,\infty)$ denote the probability density, obtained by kernel density estimation [52], of the univariate distribution of sphericities Φ for aggregates in $X^{(t,T)}$ consisting of i particles, i.e., aggregates with volume $v = iv_0$.

Based on $X^{(t,T)}$, we consider the bivariate (relative) histogram $S^{(t,T)} \in [0,1]^{64\times 64}$ of aggregate volumes and sphericities. The height $S^{(t,T)}(i,j)$ of bin $(i,j) \in \{1,\ldots,64\}^2$, where i,j specifies the bin positions of volume and sphericity, respectively, is given by

$$S^{(t,T)}(i,j) = g^{(t,T)}(i) \int_{\frac{64+j-1}{128}}^{\frac{64+j}{128}} f_i^{(t,T)}(y), \mathrm{d}y, \tag{22}$$

with $g^{(t,T)}(i) = iv_0 \,\#\{(v,\Phi) \in X^{(t,T)} : v = iv_0\} \big/ \sum_{(v,\Phi)\in X^{(t,T)}} v$ denoting the mass-weighted fraction of descriptor vectors in $X^{(t,T)}$ corresponding to aggregates of volume iv_0, where # denotes cardinality. Since sphericities have been observed to take values only in the interval $[0.5,1]$, the limits of the integral in Eq. (22) are chosen such that the histogram partitions this interval into 64 equally spaced bins. A visualization of the bivariate histogram $S^{(t,T)}$ is shown in Figure 2c.

2.3.3 Network architecture

We now explain the architecture of the neural network, i.e., a high-parametric function, which will be trained to predict $S^{(t,T)}$ based on $S^{(t-1,T)}$ and T. For this, a neural network is needed that can process both a bivariate histogram and a scalar temperature T at the same time, and can output a (predicted) bivariate histogram $P^{(t,T)} \in [0,\infty)^{64\times 64}$, corresponding to the time step t.

To the best of the authors' knowledge, there are no previous studies in the literature that address the particular task considered in the present paper, and therefore, no pre-existing architectures specifically designed for this purpose. However, there are similar efforts in image sequence prediction using surrogate models in video games [45] (Atari frame prediction network), where future frames are predicted from user input, and in predicting movement in video sequences [62] (video frame prediction network), where a person's pose is predicted after a given time interval. These networks will be used in the following as baselines, referring to them as AFPN and VFPN, respectively.

Furthermore, we consider a third small autoencoder network, denoted by SAN. Note that not only this network, but also AFPN and VFPN can be considered as autoencoders [58]. However, SAN stands out because of its very small number of parameters ($\approx 14\%$ with respect to AFPN and $\approx 1\%$ with respect to VFPN), designed specifically to learn the concepts of coagulation and sintering while avoiding overfitting, thereby ensuring feasibility, i.e., computational efficiency for real-time applications. The proposed network given by the function $\mathrm{SAN}\colon [0,1]^{64\times 64} \times [0,\infty) \to [0,1]^{64\times 64}$ can be decomposed into three parts: an encoder network $E\colon [0,1]^{64\times 64} \to \mathbb{R}^{8192}$ that extracts relevant features of the input histogram (at time step $t-1$), a fully connected network $F\colon \mathbb{R}^{8192} \times [0,\infty) \to \mathbb{R}^{128}$ that incorporates the temperature T into the features, and a decoder network $D\colon \mathbb{R}^{128} \to [0,1]^{64\times 64}$ that spatially expands these features to a bivariate histogram, predicting the histogram at time step t.Thus, for a histogram $S^{(t-1,T)}$ and a temperature T, the prediction $P^{(t,T)}$ of $S^{(t,T)}$ is given by

$$P^{(t,T)} = \mathrm{SAN}(S^{(t-1,T)}, T) = D\left(F\left(E(S^{(t-1,T)}), T\right)\right). \tag{23}$$

A schematic representation of the network architecture of SAN is shown in Figure 3. A more detailed description of this network can be found in C.

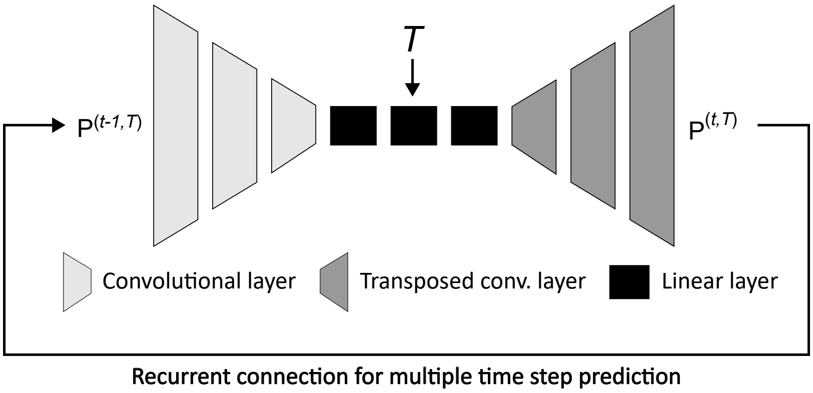


Figure 3: Architecture of the surrogate model SAN.

Recall that the networks AFPN and VFPN have a much larger number of parameters than SAN, especially VFPN. The numbers of trainable parameters per network are given in Table 2. For more details on the architecture of these two networks, see [45, 62].

In order to ensure that the outputs of the networks AFPN, VFPN, and SAN are proper bivariate histograms of relative frequencies, i.e., all entries are non-negative and sum up to one, normalization is implicitly applied to the network output. More precisely, every output matrix $O \in \mathbb{R}^{64\times 64}$ is transformed as follows:

$$O \mapsto \frac{\sigma(O)}{\sum(\sigma(O))}, \tag{24}$$

where $\sigma\colon \mathbb{R} \to [0,1]$ denotes the sigmoid activation function. Note that in Eq. (24) the activation function σ is evaluated on a matrix, where the scalar activation is evaluated on each entry of the matrix, resulting in a new matrix $\sigma(O)$ given by $\sigma(O)(i,j) = \frac{1}{1+e^{-O(i,j)}}$ for any $i,j \in \{1,\dots,64\}$ and the summation $\sum$ in Eq. (24) extends over all entries $O_{i,j}$ of O. Thereby, the sigmoid function ensures non-negativity of the bin heights, whereas the denominator in Eq. (24) ensures that the sum of all entries equals one.

2.3.4 Training procedure

To compare the similarity of histograms, especially those predicted by (untrained) network models and those derived from the MC simulations, and thus to be able to train the networks, i.e. to fit their parameters, a loss function that quantifies the dissimilarity of histograms is needed. Furthermore, to enable efficient network training, the differentiability of this loss function is required.

Let Q and R denote histograms of descriptors, which can be either univariate (1D) or bivariate (2D). In the univariate case we consider histograms of the form $Q, R \in [0,1]^{64}$, while in the bivariate case, we have $Q, R \in [0,1]^{64\times 64}$. Furthermore, let $Q(i)$ and $R(i)$ denote the height of the bin specified by i. Note that in the case of univariate histograms, $i \in \{1, \dots, 64\}$ is a scalar value, while in the case of bivariate histograms we have $i \in \{1, \dots, 64\}^2$, i.e., i is a two-dimensional vector of indices. However, for simplicity, this distinction will not be further mentioned if the context is clear.

The Kullback-Leibler divergence loss $D_{\mathrm{KL}}(Q, R)$ of Q, R is defined as

$$D_{\mathrm{KL}}(Q, R) = \sum_i Q(i) \log \left(\frac{Q(i)}{R(i)} \right), \tag{25}$$

where the sum in Eq. (25) extends over all feasible values of i, i.e. $\{1, \dots, 64\}$ or $\{1, \dots, 64\}^2$. Note that in the context of neural networks, the Kullback-Leibler divergence is frequently used to measure the difference between predicted probability distributions and ground truth probability distributions [26]. To avoid numerical issues such as taking the logarithm of zero, in practical applications of this similarity measure, a small constant $\varepsilon = 10^{-10}$ is added to $Q(i)$ and $R(i)$, modifying the definition given in Eq. (25) to

$$D_{\mathrm{KL}}(Q, R) = \sum_i Q(i) \log \left(\frac{Q(i) + \varepsilon}{R(i) + \varepsilon} \right). \tag{26}$$

The latter formula will be used in the following to evaluate the prediction quality of the networks.

The histogram $S^{(t,T)}$ resulting from the MC simulation and the histogram $P^{(t,T)}$ resulting from the network prediction are compared in three steps: First, the univariate (marginal) histograms $S_v^{(t,T)}, P_v^{(t,T)}$ of volume v are compared to each other. Second, the univariate histograms $S_\Phi^{(t,T)}, P_\Phi^{(t,T)}$ of sphericity Φ are compared, and third, the bivariate histograms $S^{(t,T)}$ and $P^{(t,T)}$ of (v, Φ) are compared. Thus, the total loss $L(S^{(t,T)}, P^{(t,T)})$ is given by

$$L(S^{(t,T)}, P^{(t,T)}) = D_{\mathrm{KL}}(S_v^{(t,T)}, P_v^{(t,T)}) + D_{\mathrm{KL}}(S_\Phi^{(t,T)}, P_\Phi^{(t,T)})) + D_{\mathrm{KL}}(S^{(t,T)}, P^{(t,T)}), \tag{27}$$

where $Q_v \in [0,1]^{64}$ and $Q_\Phi \in [0,1]^{64}$ for $Q \in \{S^{(t,T)}, P^{(t,T)}\}$ denote the marginal histograms corresponding to v and Φ, respectively, which are given by

$$Q_{\mathrm{v}}(i) = \sum_{j=1}^{64} Q(i,j) \qquad \text{and} \qquad Q_\Phi(j) = \sum_{i=1}^{64} Q(i,j), \tag{28}$$

for any $i, j \in \{1, \dots, 64\}$. Based on minimizing the loss given in Eq. (27), the neural networks' trainable parameters are updated using an Adam optimizer [6] with a learning rate of 0.0001, a batch size of 32, and 20 000 epochs.

The intention behind the construction of the loss L given in Eq. (27) is to improve the convergence of the trained network. That is, predicting marginal distributions is a much simpler task than predicting the whole bivariate distribution. After the network is able to predict the univariate distributions well, it will focus on the next, more complicated task, predicting the bivariate distribution, resulting in faster and more stable convergence.

3 Results and discussion

This section presents results, which have been obtained for the evaluation of the MC framework, analyzes the distribution dynamics and attainment of self-preserving size distributions, and evaluates the performance of the CNN-based surrogate model for predicting particle formation processes.

3.1 Evaluation of the Monte Carlo framework

The evaluation of the MC framework considered in this paper is performed in two steps. Initially, pure coagulation is compared with a simple fixed-node method. Second, the combination of coagulation and sintering is compared with bivariate models from the literature. To ensure generality in the presented data, the temporal evolution is given in terms of dimensionless time τ, which is defined for the free-molecular regime [61] as

$$\tau = \left(\frac{6k_BT}{\rho_\mathrm{p}}\right)^{1/2}\left(\frac{3}{4\pi}\right)^{1/3} v_0^{1/6} N_0 t, \tag{29}$$

where N_0 and v_0 denote the initial number and volume of primary particles, respectively, and k_B, ρ_p and T are the quantities introduced in Section 2.1. All MC simulations considered in this chapter are carried out using the parameter set in Table 1 row 1. Exceptions regarding the sintering rate are noted for the edge cases of compact spheres and fractal-like agglomerates.

3.1.1 Univariate evaluation

In a first step, the dynamics of coagulation is benchmarked using the fixed-node sectional method proposed in [46]. It consists of a simplified numerical algorithm for solving the PBE where particle sizes are represented only at discrete nodes on a logarithmic scale, which simplifies computation. The model initializes primary particles with size v_0 at a single node and then redistributes the particle volumes dynamically according to the coagulation kernel given in Eq. (3). The fixed-node method is solved for the two extreme cases of pure agglomeration ($D_\mathrm{s} = 3.0$) and full coalescence ($D_\mathrm{s} = 2.0$) of Si particles. The MC framework is compared to the solution of the fixed-node method by setting the characteristic sintering time τ_s to the constant values of $10^{16}s$ and $10^{-9}s$, respectively, which effectively correspond to the extreme cases stated above [57].

In the fixed-node method of [46] the volume grid is defined for all nodes i by $v_i = v_0\, q^i$. Grid refinement with the spacing factor q is assessed using the convergence of the geometric standard deviation $\sigma_{\mathrm{g},d_\mathrm{v}}$ of the number-weighted self-preserving size distribution (SPSD) of the particle volume-equivalent diameter d_v. The numerical solution is deemed converged when $|\sigma_{\mathrm{g},d_\mathrm{v}} - \sigma^\mathrm{ref}_{\mathrm{g},d_\mathrm{v}}|/\sigma^\mathrm{ref}_{\mathrm{g},d_\mathrm{v}} < 1$, where the reference $\sigma^\mathrm{ref}_{\mathrm{g},d_\mathrm{v}}$ is determined using a highly resolved grid ($\approx$ 20 nodes per decade). In addition, close agreement of $\sigma_{\mathrm{g},d_\mathrm{v}}$ with the values in the literature ($\sigma_{\mathrm{g},d_\mathrm{v}} = 1.46$ [11]) is required. The above stated criteria are met for $q < 1.4$, which corresponds to approximately 7 nodes per decade. The number of nodes is chosen so that the PSD is fully contained in the volume grid.

The temporal evolution of the normalized number concentration N/N_0, defined as the concentration of aggregates $N = \lim_{v\to\infty} N(v,\tau)$ at time τ divided by the initial concentration of primary particles N_0, the volume-equivalent geometric mean diameter $d_{\mathrm{v,g}}$ (Eqs. (14) and (19)) and the geometric standard deviation $\sigma_{\mathrm{g},d_\mathrm{v}}$ (Eqs. (14) and (20)), obtained from the MC framework described in Section 2.2, closely follows the fixed-node method for pure coagulation. The maximum deviation between the values produced by the two methods is at most 1%, providing a robust benchmark for the MC framework; see Figure 4.

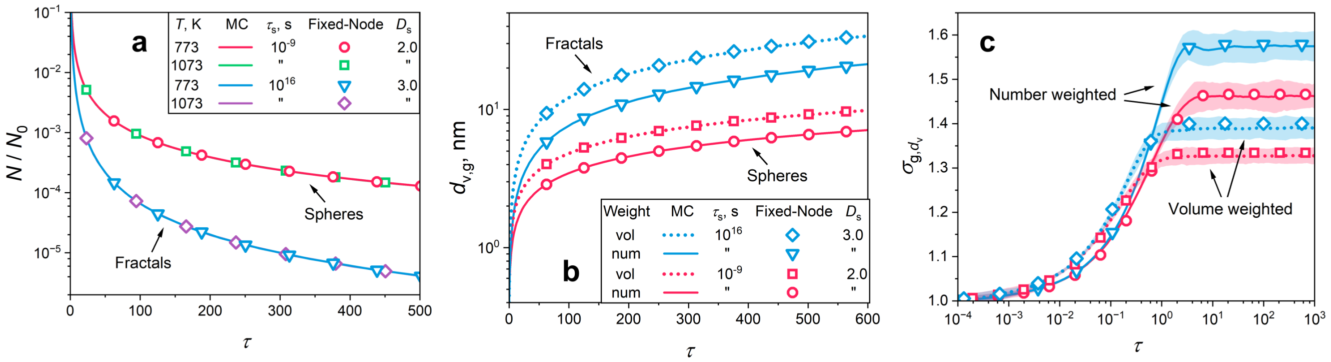


Figure 4: Temporal evolution of normalized number concentration N/N_0 (a), volume-equivalent geometric mean diameter $d_{\mathrm{v,g}}$(b), and geometric standard deviation $\sigma_{\mathrm{g},d_{\mathrm{v}}}$ (c) for comparison of the results obtained by the MC framework considered in this paper (lines) and the fixed-node benchmark (symbols) for coagulation of Si fractal-like agglomerates ($D_{\mathrm{s}} = 3.0$) and perfect spheres ($D_{\mathrm{s}} = 2.0$). Figures 4b and 4c display data generated for $T = 773\,\mathrm{K}$. The shaded region visible in Figure 4c represents twice the standard deviation of $\sigma_{\mathrm{g},d_{\mathrm{v}}}$ between all N_{Sims}.

Elevated temperatures slightly increase the collision rate of particles. In the dimensionless time domain τ, this difference is compensated (Figure 4a). However, irregular particle shapes (such as those seen in fractal-like agglomerates), on the other hand, strongly facilitate particle growth, yielding bigger particles and a fast decay in total concentration. The distributional descriptors $d_{\mathrm{v,g}}$ and $\sigma_{\mathrm{g},d_{\mathrm{v}}}$ are only shown for the single temperature value of 773 K (Figures 4b and 4c) due to temperature insensitivity.

The literature reports values of the geometric standard deviation $\sigma_{\mathrm{g},d_{\mathrm{v}}}$ characterizing SPSDs for $D_{\mathrm{f}} = 3$ (compact spheres) and $D_{\mathrm{f}} = 1.8$ (fractal-like agglomerates) [11]. For compact spheres, the MC framework converges to a $\sigma_{\mathrm{g},d_{\mathrm{v}}}$ of 1.46 and 1.33, for a number-weighted and volume-weighted distribution, respectively, consistent with the literature [11]. Fractal-like agglomerates reach a broader distribution with $\sigma_{\mathrm{g},d_{\mathrm{v}}}$ of 1.57 and 1.39. In comparison, the literature reports $\sigma_{\mathrm{g},d_{\mathrm{v}}}$ values of 1.63 and 1.42 [11]. The discrepancy is small and may be attributed to differences in the underlying coagulation kernel formulations. The SPSD is reached when the particles have grown to about three times their initial diameter $d_{\mathrm{v,g}}$, consistent with a common rule of thumb [7]. Thereby, a geometric mean number of primary particles of $n_{\mathrm{p,g}} = 15$ is attained. This is consistent with [19] and also indicates the convergence of the fractal dimension at that point in time [29]. The time delay to reach the SPSD is often given in a dimensionless form equivalent to the definition in Eq. (29) [60]. Fractal-like agglomerates reach the SPSD faster than spheres at dimensionless times of about 3.2 (for $D_{\mathrm{f}} = 2$) and 4.3 (for $D_{\mathrm{f}} = 3$) [61] respectively, which is consistent with the findings of this work in Figure 4c.

3.1.2 Bivariate evaluation

To our knowledge, there is no established benchmark for the evolution of surface area during sintering. Therefore, the present MC framework is evaluated by comparison with the (precise) bivariate discrete-sectional model of [41], which is based on [69]. Since the latter involves a non-instantaneous precursor release, our MC framework is compared to the approach proposed in [41], which describes a slightly simplified version of [69], designed to reduce computational effort. In particular, their implementation employs a coagulation kernel β given in [51], which is applicable throughout the entire Knudsen regime but assumes a constant D_{f} a priori. For further evaluation, the monodisperse model presented in [35], which simplifies the coagulation kernel by assuming all particles to be identical in size (v and a), is included in the comparison. Finally, the stochastic model considered in [42], which implements the coagulation kernel described in the present work (Eq. (3)), is also used for comparison. The material parameters for Si and the simulation settings are consistent with those of [41] and are listed in the first row of Table 1.

The results provided by our MC framework are directly compared with the temporal evolution data of normalized concentration N/N_0, normalized, number-weighted average volume $\bar{v}/v_0$, surface area-equivalent primary particle diameter $\bar{d_\mathrm{p}}$, and number of primary particles per aggregate $\bar{n_\mathrm{p}}$ (Figure 5). The results obtained are compared with the findings of [41] (see Figure 6 in [41] including the monodisperse model of [35]), and [42] (see Figure 1 in [42]) for Si aggregates under isothermal conditions. In accordance with the literature, number averages (computed using Eq. (18)) are used instead of geometric averages. To enhance comparability, the coagulation kernel of [51] is additionally implemented in our MC framework under the assumption of $D_\mathrm{f} = 1.8$.

Good agreement is found for all methods. At a temperature of 1073 K, the coalescence limit is reached, as indicated by a constant $\bar{n_\mathrm{p}} = 1$ (Figure 5d). The MC framework is closely aligned with the 2D sectional model for normalized concentration (Figure 5a) and volume (Figure 5b). For d_p (Figure 5c), the results match the monodisperse model more closely, although d_p is generally insensitive to the choice of the underlying model. However, the monodisperse model predicts lower collision rates due to its simplifying assumptions, resulting in higher concentrations and smaller volumes.

The deviations between models are more pronounced at 773 K, where aggregates are formed. The influence of different coagulation kernels is evident, with significant discrepancies observed between the MC framework and the sectional models, particularly in the evolution of normalized concentration N/N_0 and normalized average volume $\bar{v}/v_0$. However, the implementation of an identical kernel yields strong agreement between these methods, evaluating the approach and attributing the deviations solely to differences in the underlying models. In [41] a constant fractal dimension is assumed, independent of temperature and aggregate size, whereas the present model dynamically determines the surface fractal dimension $D_\mathrm{s}(v, a)$ for each individual particle. This dynamic approach reduces the average collision cross section of the aggregate population at elevated temperatures. In general, close agreement is observed between the MC simulations and the stochastic model of [42], which employs the same coagulation kernel as given in Eq. (3). The monodisperse model of [35] shows close agreement with the present MC simulations, despite its simplified dynamics and differing coagulation kernel. This consistency presumably arises from two opposing effects that effectively counteract each other. As noted previously, the use of a constant fractal dimension D_f in the coagulation kernel leads to an overestimation of the overall collision frequency compared to the variable $D_\mathrm{s}(v, a)$. Moreover, the monodisperse assumption causes a reduction in collision frequency compared to polydisperse systems [25].

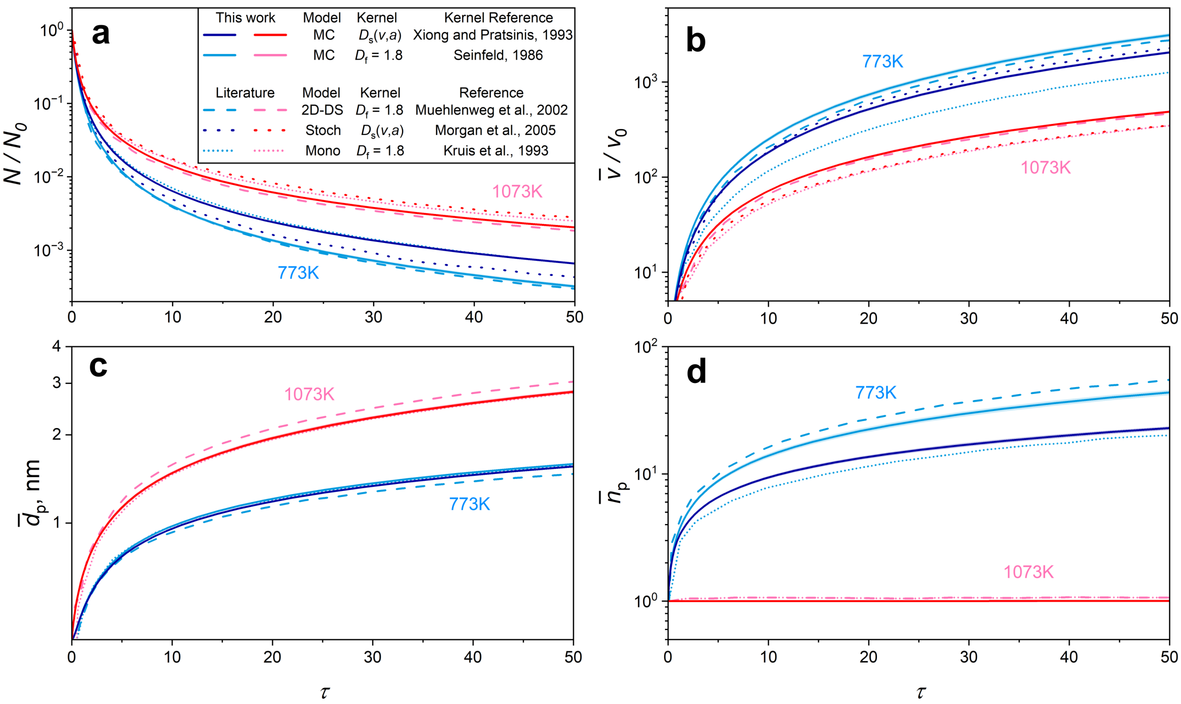
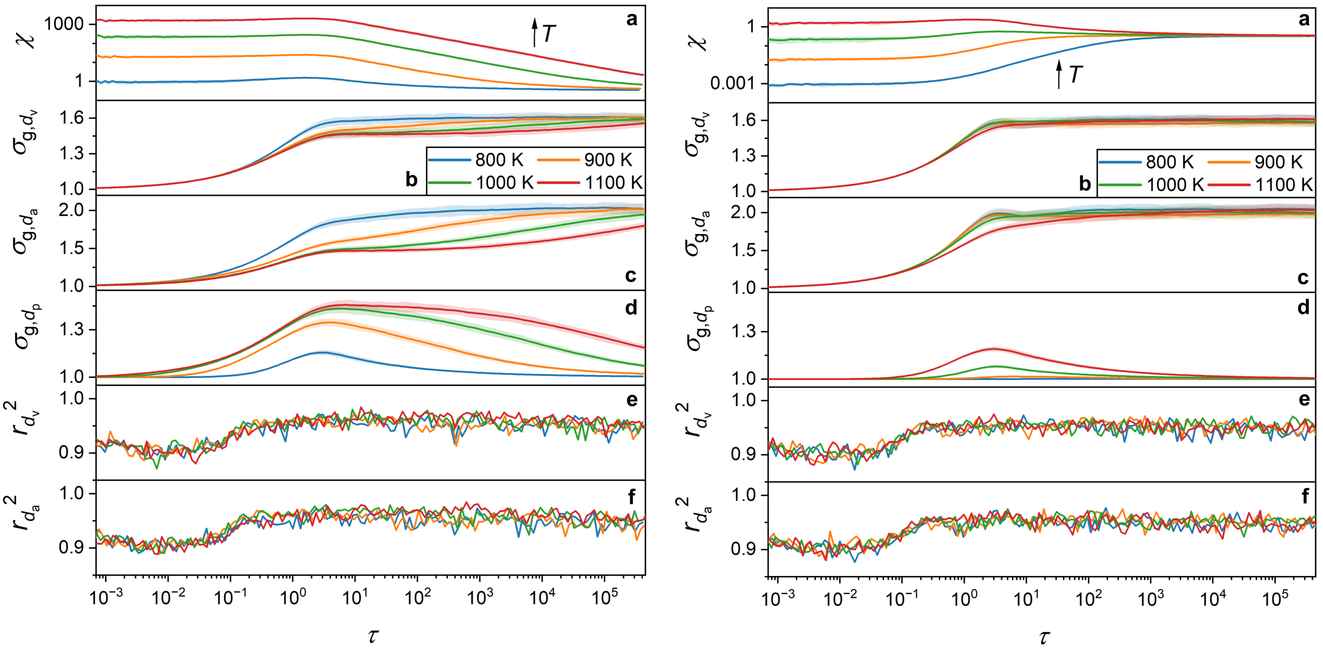


Figure 5: Temporal evolution of normalized number concentration (a), normalized volume (b), primary particle diameter (c) and number of primary particles per aggregate (d) of Si aggregates for evaluation of the present bivariate MC framework (dark solid). Direct comparison to the 2D discrete-sectional model (2D-DS) presented in [41] (dashed), the stochastic model (Stoch) of [42] (dotted) and the monodisperse model of [35] (short dotted) is shown. The coagulation kernel of the present framework with dependency on variable $D_s(v, a)$ (given in Eq. (3) (dark lines) by [69]) is also used in the stochastic model [42] but data is only available for the evolution of concentration and volume. The other two comparative models employ another coagulation kernel (given by [51]) dependent on a constant $D_f = 1.8$ (light lines), which is also implemented in the present MC framework (light solid) for a more accurate comparison to those models. Two different cases for $T = 773\,\mathrm{K}$ (blue) and $T = 1073\,\mathrm{K}$ (red) are given.

3.2 Self-preserving size distributions of volume and surface area

Under Brownian coagulation, the number distribution of volume-equivalent diameters d_v evolves toward an asymptotic self-preserving size distribution (SPSD) [36, 15]. In this regime, the distribution attains a self-similar form, such that its shape remains time-invariant when normalized by the total particle concentration N and scaled by the geometric mean volume-equivalent diameter $d_{v,g}$. This concept is widely accepted and frequently used, for example, in conjunction with monodisperse models to predict particle size distribution formation dynamics [20]. In contrast, surface area-based SPSDs have received comparatively little attention. Joint volume and surface area distributions have been proposed and studied for both the continuum [56] and the free-molecular regimes [67]. These studies have been carried out under the assumption of constant fractal dimensions D_f and constant characteristic sintering times τ_s, thereby neglecting the variability of D_f in distributed systems and the strong size dependence of the primary particle size on τ_s. The present study focuses on Fe aggregates, examining the conditions under which SPSDs are attained for both volume v and surface area a and thus, providing a framework for improved, simplified prediction of the formation dynamics of aggregate systems. Additionally, the correlation of PSD shapes for volume-equivalent d_v and surface-area equivalent diameters d_a with an ideal lognormal distribution is investigated. The settings and material parameters used for the evaluations in this chapter are given in Table 1 row 2.

The dimensionless temporal evolution of the geometric standard deviations $\sigma_{\mathrm{g},d_\mathrm{v}}$, $\sigma_{\mathrm{g},d_\mathrm{a}}$, and σ_{g,d_p} based on the volume-equivalent diameter d_v, surface area-equivalent diameter d_a, and the primary particle diameter d_p respectively, is shown in Figures 6.I and 6.II for $N_0 = 8 \cdot 10^{19}$ 1/m^3 and $N_0 = 8 \cdot 10^{22}$ 1/m^3. Simulations were conducted at various temperatures. As expected, the ratio of characteristic times, χ (see Eq. (15)), increases with temperature due to accelerated sintering (see Figures 6.Ia and 6.IIa) and is strongly influenced by particle concentration. Figure 6.I corresponds to sintering-dominated dynamics ($\chi \gg 1$), while Figure 6.II corresponds to coagulation-dominated growth ($\chi \ll 1$). Over time, χ converges to intermediate values between 0.1 and 1, reflecting concurrent increases in both characteristic coagulation τ_c (due to reduction of N) and characteristic sintering time τ_s (due to growth in d_p). These intermediate values of χ are typical for aggregate formation [7].



(I) $N_0 = 8 \cdot 10^{19}$ 1/m^3

(II) $N_0 = 8 \cdot 10^{22}$ 1/m^3

Figure 6: Temporal evolution of the ratio of characteristic timescales χ, geometric standard deviations $\sigma_{\mathrm{g},d_\mathrm{v}}$, $\sigma_{\mathrm{g},d_\mathrm{a}}$, and σ_{g,d_p}, as well as the Pearson correlation $r^2_{d_\mathrm{v}}$ and $r^2_{d_\mathrm{a}}$ of d_v and d_a distributions to their corresponding lognormal probability density functions. Results are shown for initially low (I) and high concentrations (II) of Fe primary particles, which result in initially large $\chi \gg 1$ and small $\chi \ll 1$ respectively.

The parameter χ has substantial impact on the evolution of the distributions and attainment of constant $\sigma_{\mathrm{g},d_\mathrm{v}}$ and $\sigma_{\mathrm{g},d_\mathrm{a}}$. The classical SPSD theory predicts a time delay for the d_v number distribution between 3 and 4.3 [61], depending on D_f in the free-molecular regime. As shown in Figures 6.Ib and 6.IIb, such rapid asymptotic behavior of $\sigma_{\mathrm{g},d_\mathrm{v}}$ occurs only for extreme values of χ (Figures 6.Ia and 6.IIa). For example, Figures 6.Ib-6.Id display rapid attainment of $\sigma_{g,d_\mathrm{v}} = \sigma_{g,d_\mathrm{a}} = \sigma_{g,d_\mathrm{p}} = 1.46$ associated with a perfect SPSD for spherical particles near the coalescence limit at 1000 K, which stays approximately constant until $\tau \sim 1000$ when χ decreases. Similarly, Figures 6.IIb and 6.IIc show $\sigma_{g,d_{\mathrm{v}}} = 1.57$ and $\sigma_{g,d_{\mathrm{a}}} = 1.98$, which are quickly attained at lower temperatures (800 K − 900 K), where $\chi \ll 1$. However, in intermediate cases of χ, achieving constant $\sigma_{\mathrm{g},d_{\mathrm{a}}}$ takes significantly longer and slightly longer for $\sigma_{\mathrm{g},d_{\mathrm{a}}}$, which coincides with convergence to a monodisperse primary particle size in the aggregate population. Although the rule of thumb suggests that d_{v}-based SPSD is reached when the initial diameter triples [7], this does not hold for intermediate cases of χ. For example, at low initial concentration (and 900 K), constant $\sigma_{\mathrm{g},d_{\mathrm{v}}}$ occur only after a 400-fold increase in diameter at a dimensionless time of approximately 50 000.

Next, we discuss how well the distribution of d_{a} and d_{v} can be represented by a lognormal distribution, as this assumption is frequently postulated [14] and applied in numerical prediction methods, such as the method of moments [65], to facilitate the modeling of particle dynamics. Despite its widespread use, the validity of the lognormal assumption is rarely examined or quantified. For example, deviations from lognormal distribution shapes have been reported in [70], where fractal collision models indicate reduced collision rates between small and large aggregates, broadening the distributions and amplifying deviations from the lognormal shape. In the present study, the similarity to a fitted lognormal distribution is quantified using the square of the Pearson correlation coefficient r^2. Here, r^2 is employed as a quantitative measure of distribution shape similarity, rather than as a formal statistical goodness-of-fit metric, as it provides a simple and scale-independent basis for comparing the temporal evolution of distribution shapes. It quantifies the similarity between two empirical cumulative distribution functions (CDFs) F_{MC} and $F_{\mathrm{lognormal}}$, as

$$r^2(F_{\mathrm{MC}}, F_{\mathrm{lognormal}}) = \left(\frac{\mathrm{Cov}(F_{\mathrm{MC}}, F_{\mathrm{lognormal}})}{\sigma_{F_{\mathrm{MC}}} \sigma_{F_{\mathrm{lognormal}}}} \right)^2, \tag{30}$$

where $\mathrm{Cov}(F_{\mathrm{MC}}, F_{\mathrm{lognormal}})$ is the empirical covariance of F_{MC} and $F_{\mathrm{lognormal}}$, and $\sigma_{F_{\mathrm{MC}}}$, $\sigma_{F_{\mathrm{lognormal}}}$ are the empirical standard deviations of F_{MC} and $F_{\mathrm{lognormal}}$. At each time point, a lognormal distribution is fitted to the weighted MC particle data, and a synthetic sample of size N_{SP} is drawn from the fitted distribution. The empirical CDFs F_{MC} and $F_{\mathrm{lognormal}}$ are then evaluated on a common logarithmically spaced grid with size $N_{\mathrm{grid}} = 500$ with $d_{\mathrm{v},1}, \ldots, d_{\mathrm{v},N_{\mathrm{grid}}} \in [d_{\mathrm{v,min}}, d_{\mathrm{v,max}}]$ and $d_{\mathrm{a},1}, \ldots, d_{\mathrm{a},N_{\mathrm{grid}}} \in [d_{\mathrm{a,min}}, d_{\mathrm{a,max}}]$, for d_{v} and d_{a} distribution data respectively.

The evolution of the correlation between both d_{v} and d_{a} distributions to ideal lognormal shape is largely independent of temperature and initial concentration once time is expressed in the dimensionless form (Eq. (29)) as seen for example in Figures 6.Ie and 6.IIe. The temporal evolutions of $r^2_{d_{\mathrm{a}}}$ and $r^2_{d_{\mathrm{v}}}$ also match well, allowing to judge the influence of sintering on the distribution shape as negligible. As coagulation sets in, size-dependent collision efficiencies and the emergence of aggregates lead to asymmetric broadening and driving r^2 slightly down to $r^2 \approx 0.9 \pm 0.01$, consistent with the qualitative findings of [70]. As the characteristic coagulation time proceeds to increase, due to the falling concentration, the particle system converges to an asymptotic distribution shape, which is reflected by an increase in r^2 and stabilization at $r^2 \approx 0.95 \pm 0.02$ at $\tau \approx 0.2$ independent of $\sigma_{\mathrm{g},d_{\mathrm{v}}}$, $\sigma_{\mathrm{g},d_{\mathrm{a}}}$ or $\sigma_{\mathrm{g},d_{\mathrm{p}}}$ convergence. The $r^2_{d_{\mathrm{v}}}$ for a perfect d_{v}-based SPSD can therefore also be associated with a value of approximately 0.95 displaying a slight skewness compared to an ideal lognormal distribution. The findings suggest that the lognormal assumption is approximately accurate for fully developed distributions within the present parameter range but slightly erroneous in early growth stages.

The key findings of this chapter are summarized as follows. The SPSD of d_{a} and d_{v} are attained under three conditions:

1. Sufficiently large $\chi >> 1$ (near the coalescence limit), leading to spherical particles.

2. Sufficiently small $\chi << 1$ (negligible sintering), leading to agglomerates.

3. Aggregates with uniform d_p throughout the population, which requires significantly more time than the first two cases, and the classical SPSD theory suggests.

These results demonstrate that a d_a-based SPSD coincides with the classical d_v-based SPSD in the present observations. Furthermore, independent of the attainment of SPSDs, lognormal distribution shapes for both d_v and d_a distributions are reached and remain stable over time. These insights simplify the prediction of bivariate aggregate system distributions, particularly in conjunction with monodisperse formation models [35] if χ can be estimated.

Note that the presented investigations consider isothermal conditions, which are often not applicable in real-world scenarios and synthesis methods such as spark ablation, arc-discharge, or flame spray pyrolysis, but they apply well to, for example, hot-wall reactors. While both d_v and d_a size distributions attain self-preserving forms, this does not imply self-preservation of the corresponding joint distribution, as correlations between volume and surface area may evolve over time.

3.3 Evaluation of the surrogate model

During the training process, the networks AFPN, VFPN and SAN introduced in Section 2.3.3 were only trained to predict the distribution of the bivariate descriptor vector consisting of the volume and surface area of aggregates for the next time step. However, in applications, it is desired to use the surrogate model for predicting the entire time series of histograms. In other words, the networks are used to predict a time series of histograms over 1 second, resulting in a series of 101 bivariate histograms, all at a constant temperature T. More precisely, for a network $f \in \{\mathrm{AFPN}, \mathrm{VFPN}, \mathrm{SAN}\}$, the series of histograms $P^{(0,T)}, \ldots, P^{(100,T)}$ is computed, where

$$P^{(t,T)} = \begin{cases} P^{(0,T)}, & \text{if } t = 0, \\ f(P^{(t-1,T)}, T), & \text{else,} \end{cases} \tag{31}$$

for $t \in \{0, \ldots, 100\}$. Note that $P^{(0,T)}$ is the histogram of monodisperse primary particles, i.e., aggregates consisting of one perfectly spherical primary particle.

The time series $P^{(0,T)}, \ldots, P^{(100,T)}$ given in Eq.(31) is computed for all temperatures T considered in the evaluation dataset and for all three network architectures AFPN, VFPN and SAN. In Table 2, the mean time necessary for the computation of a $P^{(0,T)}, \ldots, P^{(100,T)}$ per network is displayed. It can be observed that the surrogate models terminate significantly faster than the MC simulation and, thus, are more suitable for autonomous process control. Thereby, SAN shows the highest computational efficiency.

To quantify the quality of the network predictions, five different characteristics are considered. Namely, the Bhattacharyya distance D_B, the chi-square distance D_{χ^2}, the symmetric Kullback-Leibler divergence D_SKL, the L_1-distance, and the L_2-distance [8]. For two histograms $Q, R \in$

$[0, 1]^{64\times 64}$, these characteristics are given by

$$D_B(Q, R) = -\ln\left(\sum_i \sqrt{Q(i)R(i)}\right), \tag{32}$$

$$D_{\chi^2}(Q, R) = \sum_i \frac{(Q(i) - R(i))^2}{Q(i) + R(i)}, \tag{33}$$

$$D_{\mathrm{SKL}}(Q, R) = \frac{1}{2}(D_{\mathrm{KL}}(Q, R) + D_{\mathrm{KL}}(R, Q)), \tag{34}$$

$$L_1(Q, R) = \sum_i |(Q(i) - R(i))|, \tag{35}$$

$$L_2(Q, R) = \sqrt{\sum_i (Q(i) - R(i))^2}, \tag{36}$$

where D_{KL} is given by Eq. (25) and i extends over all values of $i \in \{1, \ldots, 64\}^2$.

In Figure 7, the prediction qualities of the different network architectures and some MC-based reference (colors) are shown for different distance measures (individual plots) over time (x-axis). The shaded areas indicate the standard deviations across simulations at different temperatures; their narrow widths suggest that predictive performance varies weakly with temperature. All considered measures show the same qualitative trend. Overall, SAN seems to outperform both AFPN and VFPN. To evaluate this more quantitatively, the mean accuracy over all time steps and temperatures is shown in Table 2. The low accuracy of VFPN could be due to its high number of parameters whereby it tends to overfit, creating artifacts in its time series predictions, as can be seen in Figure 7 for later time steps.

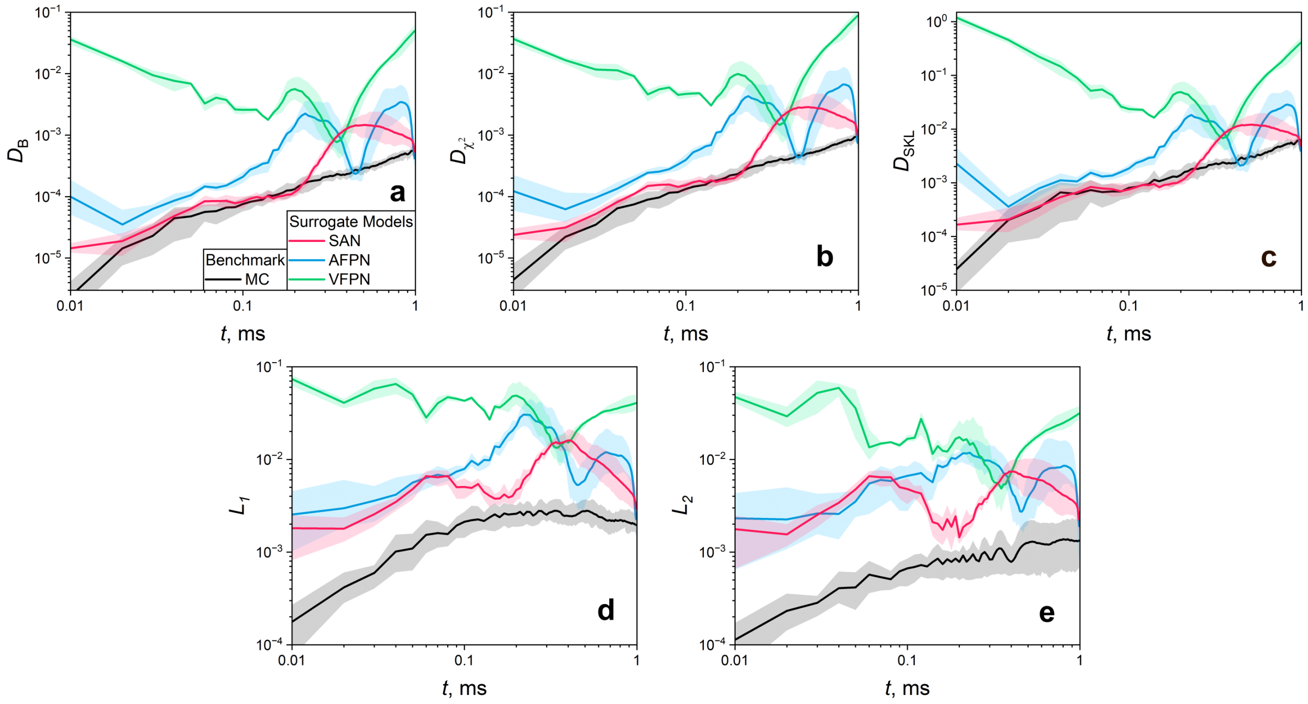
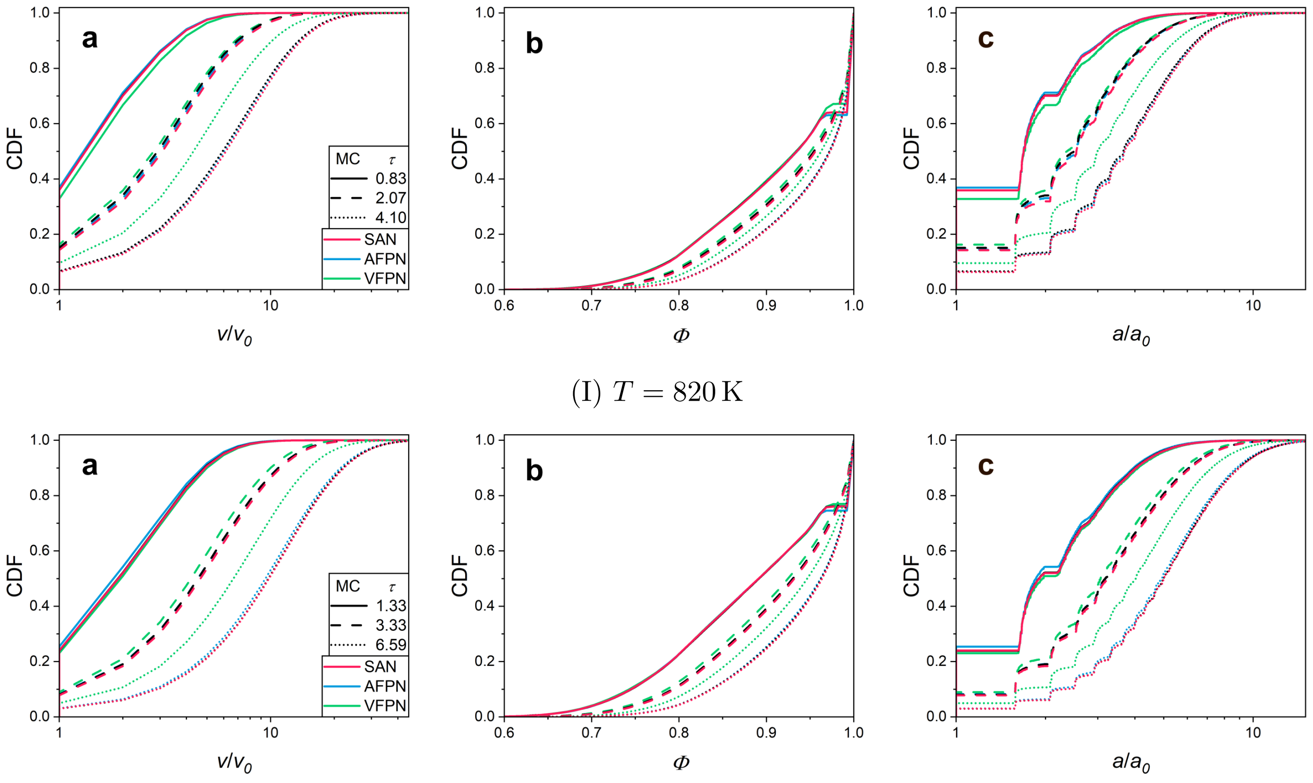


Figure 7: Comparison of network architectures: The graphs show the distance between predicted and simulated histograms. The x-axis indicates the time span into the future being predicted; the color encodes the prediction method. The bold line represents the mean distance across all temperatures in the training dataset at a given time span. The transparent shaded region shows the 0.1 and 0.9 quantiles of the distances. Green, blue, and red correspond to the network predictions. The black line represents a second independent MC simulation of 10 000 particles, used to quantify the uncertainty present in the MC simulation. This line provides a lower bound on the prediction accuracy. Each plot corresponds to a different distance measure, see Eqs. (32)–(36).

network	parameters	ct	D_B	D_{χ^2}	D_{SKL}	L_1	L_2
MC	-	1 883s	2.893e-04	7.835e-04	3.900e-03	6.157e-03	7.205e-03
SAN	2 442 368	0.118s	8.386e-04	2.455e-03	9.520e-03	1.754e-02	2.211e-02
AFPN	13 206 349	0.207s	1.512e-03	4.441e-03	1.698e-02	2.847e-02	3.530e-02
VFPN	177 536 640	0.315s	1.350e-02	3.785e-02	1.635e-01	1.961e-01	2.152e-01

Table 2: Comparison of network architectures: As a baseline, the MC framework is displayed. The computer time (ct) corresponds to the prediction of a series of 100 bivariate histograms. The computer time is given as a arithmetic mean over all temperature T of the validation data set. The losses correspond to the mean loss over all time series predictions for all temperatures in the evaluation data split, see Figure 7.

In Figure 8, we show the cumulative distribution functions (CDFs) of the predicted marginal distributions Φ, v, and a (see Eq. (21)) for several time steps and temperatures, together with a reference from the MC simulation. A high similarity between all the CDFs of SAN, AFPN and the ground truth data for all time steps and temperatures can be observed.



(I) $T = 820\,\mathrm{K}$

(II) $T = 2120\,\mathrm{K}$

Figure 8: Cumulative distribution functions of (a) volume, (b) sphericities and (c) surface areas of time series predictions for all network architectures (decoded by colors), time steps given on the x-axis, and two different temperatures: (I) $T = 820\,\mathrm{K}$, (II) $T = 2120\,\mathrm{K}$.

Beyond the accuracy of these surrogate models, prediction speed is a crucial factor. If the model is not significantly faster, it cannot be effectively utilized in autonomous process control. Table 2 presents the prediction speeds for all network architectures, as well as for the MC simulation. It is evident that a substantial increase in prediction speed is achieved across all network architectures, with the most significant improvement observed in the case of SAN. Additionally, predictions are performed on a GPU, enabling easy parallelization for multiple simultaneous predictions. However, whether the prediction speed is sufficient to support gradient-based temperature optimization during an arc synthesis process remains a topic for future investigation.

In order to control a process predicted by the proposed surrogate model approach, the surrogate model must be feasible in predicting temperatures T that lead to a desired particle descriptor vector distribution at a specified time step t. That is, an inverse problem should be solved. The applicability of the presented surrogate model to solve such an inverse problem is investigated next. In order to quantitatively evaluate the result, the desired particle descriptor vector distributions are chosen as $S^{100,T}$ for T in the evaluation data split. Then, for each of these desired descriptor vector distributions, an optimal temperature $\widehat{T} \in R$ is determined as the temperature that maximizes the similarity of the predicted histogram $P^{100,\widehat{T}}$ and the desired one $S^{(100,T)}$, i.e.,

$$\widehat{T} = \operatorname*{argmin}_{x>0} D_{\mathrm{KL}}(S^{(100,T)}, P^{(100,x)}), \tag{37}$$

where $P^{(100,x)}$ is given by Eq. (31). Due to the (almost everywhere) differentiability of the surrogate model (see ReLU activation function), this minimization problem can be efficiently solved using gradient descent. In Figure 9, the resulting optimized temperature per architecture can be observed. Both SAN and AFPN perform well on this task, whereas VFPN does not achieve satisfying results. The whole optimization procedure for all visible data points took 25s, 50s, and 180s, respectively.

However, for this type of inverse problem, the optimization is not limited to constant temperatures T. Instead, time-dependent temperatures $T(t)$ can also be optimized, offering a more flexible and potentially more effective solution. For simplicity and practicality, we propose to assume that the temperature gradient follows a polynomial of degree d. This assumption reduces the optimization problem to adjusting $d+1$ parameters. The resulting temperature profile function is smooth, reflecting real-world applications where abrupt changes in temperature are infeasible. However, an investigation of this approach, as well as its application in autonomous process control, will be the subject of future work.

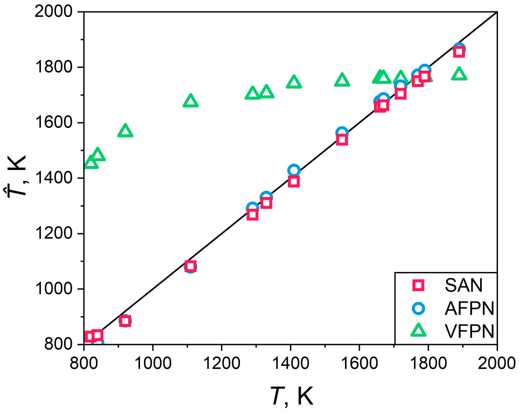


Figure 9: Predicted temperature: The predicted temperature $\widehat{T}$ is shown, which maximizes the similarity between predicted $P^{(100,\widehat{T})}$ and $S^{(100,T)}$ for all values of T of the evaluation data set. The shape and color of the points, correspond to the different considered neural network architectures.

4 Conclusion

This study presents a bivariate Monte Carlo (MC) framework optimized for computational efficiency. The framework is benchmarked against established literature models, demonstrating high agreement in the prediction of particle formation dynamics during coagulation and sintering.

To further accelerate computation and achieve real-time predictive capability, a CNN-based surrogate model is introduced. This model predicts 2D probability distributions of particle properties as a function of temperature, representing the predictions as bivariate histograms. These histograms form time series that are differentiable, enabling efficient optimization of temperature profiles for process control. The CNN demonstrates high predictive accuracy, including for temperatures not seen during training. However, a limitation is that the bins for the histograms must be fixed a priori, requiring predefined ranges for aggregate volume and surface area.

The present MC framework offers insights into the attainment of self-preserving size distributions (SPSDs). Beyond the classical volume-based SPSD, this study identifies and characterizes conditions for attaining a surface area SPSD, which is rarely observed in the literature. Both SPSDs are rapidly attained in cases of coalescence ($\sigma_{g,d_\mathrm{v}} = \sigma_{g,d_\mathrm{a}} = 1.46$) and pure agglomeration ($\sigma_{g,d_\mathrm{v}} = 1.57$ and $\sigma_{g,d_\mathrm{a}} = 1.98$), consistent with classical SPSD theory for d_v-based, number-weighted distributions. However, for aggregates in intermediate stages, the SPSDs require reaching uniform primary particle sizes across the aggregate population, which can take significantly more time. Additionally, it is observed and quantified that the d_v and d_a distribution shapes align closely with a lognormal distribution but slight deviations are observed in the early growth stages. These findings suggest the potential for simplified bivariate modeling approaches as a framework to enhance surrogate model performance in later time steps, where accuracy typically diminishes.

Future work will focus on extending the surrogate model to be trained on data with dynamic sintering times and adding another input dimension in the form of variable initial concentrations, which is necessary to capture the dynamics of precursor evaporation rates in processes such as arc-discharge synthesis. Additionally, incorporating temporal temperature gradients into the model will further enhance its applicability to real-world scenarios. Such advancements could transform arc-discharge synthesis from a semi-controlled process to a fully optimized and reliable one, reducing variability, enhancing the sustainability of nanoparticle production and laying the groundwork for closed-loop control systems capable of real-time optimization. The surrogate model could be integrated within the canonical correlation deep neural network (CCDNN) framework, which is suited for monitoring and correlation analysis, to provide robust insights into temperature-property interactions and validate predictive outcomes.

Acknowledgement

This work has been funded by the German Research Foundation (DFG), project numbers 504661005 and 504580586, as part of the priority program “Autonomous Processes in Particle Technology – Research and Testing of Concepts for Model-based Control of Particulate Processes” (SPP 2364).

References

[1] M. K. Akhtar, G. G. Lipscomb, and S. E. Pratsinis. Monte Carlo simulation of particle coagulation and sintering. *Aerosol Science and Technology*, 21:83–93, 1994.

[2] J. Appel, H. Bockhorn, and M. Wulkow. A detailed numerical study of the evolution of soot particle size distributions in laminar premixed flames. *Chemosphere*, 42:635–645, 2001.

[3] C. Artelt, H.-J. Schmid, and W. Peukert. On the impact of accessible surface and surface energy on particle formation and growth from the vapour phase. *Journal of Aerosol Science*, 36:147–172, 2005.

[4] P. Biswas and C. Y. Wu. Control of toxic metal emissions from combustors using sorbents: A review. *Journal of the Air and Waste Management Association*, 48:113–127, 1998.

[5] L. Bliek. A survey on sustainable surrogate-based optimisation. *Sustainability*, 14:3867, 2022.

[6] S. Bock, J. Goppold, and M. Weiß. An improvement of the convergence proof of the ADAM-Optimizer. *arXiv preprint arXiv:1804.10587*, 2018.

[7] B. Buesser and S. E. Pratsinis. Design of nanomaterial synthesis by aerosol processes. *Annual Review of Chemical and Biomolecular Engineering*, 3:103–127, 2012.

[8] S.-H. Cha. Comprehensive survey on distance/similarity measures between probability density functions. *International Journal on Mathematical Models and Methods in Applied Sciences*, 1:300–307, 2007.

[9] Z. Chen, K. Liang, S. X. Ding, C. Yang, T. Peng, and X. Yuan. A comparative study of deep neural network-aided canonical correlation analysis-based process monitoring and fault detection methods. *IEEE Transactions on Neural Networks and Learning Systems*, 33:6158–6171, 2022.

[10] Z. Chen, S. Mo, H. Ke, S. X. Ding, Z. Jiang, C. Yang, and W. Gui. Canonical correlation guided deep neural network. *arXiv preprint arXiv:2409.19396*, 2024.

[11] P. J. Dekkers and S. K. Friedlander. The self-preserving size distribution theory. I. Effects of the Knudsen number on aerosol agglomerate growth. *Journal of Colloid and Interface Science*, 248:295–305, 2002.

[12] M. L. Eggersdorfer and S. E. Pratsinis. Agglomerates and aggregates of nanoparticles made in the gas phase. *Advanced Powder Technology*, 25:71–90, 2014.

[13] M. Frenklach and S. J. Harris. Aerosol dynamics modeling using the method of moments. *Journal of Colloid and Interface Science*, 118:252–261, 1987.

[14] S. K. Friedlander. *Smoke, Dust, and Haze: Fundamentals of Aerosol Dynamics.* Oxford University Press, 2nd edition, 2000.

[15] S. K. Friedlander and C. S. Wang. The self-preserving particle size distribution for coagulation by brownian motion. *Journal of Colloid and Interface Science*, 22:126–132, 1966.

[16] L. Fuchs, T. Kirstein, C. Mahr, O. Furat, V. Baric, A. Rosenauer, L. Mädler, and V. Schmidt. Using convolutional neural networks for stereological characterization of 3d hetero-aggregates based on synthetic stem data. *Machine Learning: Science and Technology*, 5:025007, 2024.

[17] O. Furat, M. Wang, M. Neumann, L. Petrich, M. Weber, C. E. Krill III, and V. Schmidt. Machine learning techniques for the segmentation of tomographic image data of functional materials. *Frontiers in Materials*, 6:145, 2019.

[18] D. T. Gillespie. An exact method for numerically simulating the stochastic coalescence process in a cloud. *Journal of Atmospheric Sciences*, 32:1977–1989, 1975.

[19] E. Goudeli, M. L. Eggersdorfer, and S. E. Pratsinis. Coagulation of agglomerates consisting of polydisperse primary particles. *Langmuir*, 32:9276–9285, 2016.

[20] E. Goudeli and S. E. Pratsinis. Gas-phase manufacturing of nanoparticles: Molecular dynamics and mesoscale simulations. *Particulate Science and Technology*, 34:483–493, 2016.

[21] X. Hao, H. Zhao, Z. Xu, and C. Zheng. Population balance-Monte Carlo simulation for gas-to-particle synthesis of nanoparticles. *Aerosol Science and Technology*, 47:1125–1133, 2013.

[22] V. Harish, M. Ansari, D. Tewari, A. B. Yadav, N. Sharma, S. Bawarig, M.-L. García-Betancourt, A. Karatutlu, M. Bechelany, and A. Barhoum. Cutting-edge advances in tailoring size, shape, and functionality of nanoparticles and nanostructures: A review. *Journal of the Taiwan Institute of Chemical Engineers*, 149:105010, 2023.

[23] M. C. Heine and S. E. Pratsinis. Brownian coagulation at high concentration. *Langmuir*, 23:9882–9890, 2007.

[24] A. C. Hindmarsh, P. N. Brown, K. E. Grant, S. L. Lee, R. Serban, D. E. Shumaker, and C. S. Woodward. SUNDIALS: Suite of nonlinear and differential/algebraic equation solvers. *ACM Transactions on Mathematical Software*, 31:363–396, 2005.

[25] W. C. Hinds. *Aerosol Technology: Properties, Behavior, and Measurement of Airborne Particles*. Wiley-VCH, 1999.

[26] J. Huang, X. Sun, S. X. Ding, X. Yang, and O. K. Ersoy. Variational discriminative stacked auto-encoder: Feature representation using a prelearned discriminator, and its application to industrial process monitoring. *IEEE Transactions on Neural Networks and Learning Systems*, 36:9383–9394, 2025.

[27] N. Ibtehaz and M. S. Rahman. Multiresunet: Rethinking the u-net architecture for multimodal biomedical image segmentation. *Neural Networks*, 121:74–87, 2020.

[28] M. Jacobson and J. H. Seinfeld. Evolution of nanoparticle size and mixing state near the point of emission. *Atmospheric Environment*, 38:1839–1850, 2004.

[29] G. A. Kelesidis and S. E. Pratsinis. A perspective on gas-phase synthesis of nanomaterials: Process design, impact and outlook. *Chemical Engineering Journal*, 421:129884, 2021.

[30] T. Kirstein, L. Petrich, R. R. P. Purushottam Raj Purohit, J.-S. Micha, and V. Schmidt. Cnn-based laue spot morphology predictor for reliable crystallographic descriptor estimation. *Materials*, 16:3397, 2023.

[31] W. Koch and S. K. Friedlander. The effect of particle coalescence on the surface area of a coagulating aerosol. *Journal of Colloid and Interface Science*, 140:419–427, 1990.

[32] G. Kotalczyk and F. E. Kruis. A Monte Carlo method for the simulation of coagulation and nucleation based on weighted particles and the concepts of stochastic resolution and merging. *Journal of Computational Physics*, 340:276–296, 2017.

[33] M. Kraft. Modelling of particulate processes. *KONA Powder and Particle Journal*, 23:18–35, 2005.

[34] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.

[35] F. E. Kruis, K. A. Kusters, S. E. Pratsinis, and B. Scarlett. A simple model for the evolution of the characteristics of aggregate particles undergoing coagulation and sintering. *Aerosol Science and Technology*, 19:514–526, 1993.

[36] F. S. Lai, S. K. Friedlander, J. Pich, and G. M. Hidy. The self-preserving particle size distribution for brownian coagulation in the free-molecule regime. *Journal of Colloid and Interface Science*, 39:395–405, 1972.

[37] J. D. Landgrebe and S. E. Pratsinis. A discrete-sectional model for particulate production by gas-phase chemical reaction and aerosol coagulation in the free-molecular regime. *Journal of Colloid and Interface Science*, 139:63–86, 1990.

[38] L. Li, S. X. Ding, K. Liang, Z. Chen, and T. Xue. Control theoretically explainable application of autoencoder methods to fault detection in nonlinear dynamic systems. 2023. https://doi.org/10.48550/arXiv.2208.01291.

[39] Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou. A survey of convolutional neural networks: Analysis, applications, and prospects. *IEEE Transactions on Neural Networks and Learning Systems*, 33:6999–7019, 2021.

[40] K. McBride and K. Sundmacher. Overview of surrogate modeling in chemical process engineering. *Chemie Ingenieur Technik*, 91:228–239, 2019.

[41] H. Mühlenweg, A. Gutsch, A. Schild, and S. E. Pratsinis. Process simulation of gas-to-particle-synthesis via population balances: Investigation of three models. *Chemical Engineering Science*, 57:2305–2322, 2002.

[42] M. K. N. Morgan, C. Wells and W. Wagner. Modelling nanoparticle dynamics: Coagulation, sintering, particle inception and surface growth. *Combustion Theory and Modelling*, 9:449–461, 2005.

[43] J. Na, K. Jeon, and W. B. Lee. Toxic gas release modeling for real-time analysis using variational autoencoder with convolutional neural networks. *Chemical Engineering Science*, 181:68–78, 2018.

[44] S. Nikolopoulos, I. Kalogeris, and V. Papadopoulos. Non-intrusive surrogate modeling for parametrized time-dependent PDEs using convolutional autoencoders. *arXiv preprint arXiv:2101.05555*, 2021.

[45] J. Oh, X. Guo, H. Lee, R. L. Lewis, and S. Singh. Action-conditional video prediction using deep networks in Atari games. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28, pages 2863–2871. Curran Associates, Inc., 2015.

[46] A. Prakash, A. P. Bapat, and M. R. Zachariah. A simple numerical algorithm and software for solution of nucleation, surface growth, and coagulation problems. *Aerosol Science and Technology*, 37:892–898, 2003.

[47] T. Rosenberger, I. Skenderović, J. Sellmann, P. Wollny, A. Levish, I. Wlokas, A. Kempf, M. Winterer, and F. E. Kruis. Determining the sintering kinetics of Fe and Fe x o y - nanoparticles in a well-defined model flow reactor. *Aerosol Science and Technology*, 56:833–846, 2022.

[48] M. Sander, R. H. West, M. S. Celnik, and M. Kraft. A detailed model for the sintering of polydispersed nanoparticle agglomerates. *Aerosol Science and Technology*, 43:978–989, 2009.

[49] A. Schmidt-Ott. New approaches to in situ characterization of ultrafine agglomerates. *Journal of Aerosol Science*, 19:553–563, 1988.

[50] A. Schmidt-Ott. *Spark Ablation.* Jenny Stanford Publishing, 2019.

[51] J. H. Seinfeld. *Atmospheric Chemistry and Physics of Air Pollution.* Wiley-VCH, 1986.

[52] B. W. Silverman. *Density Estimation for Statistics and Data Analysis.* Routledge, 2018.

[53] M. Smoluchowski. Mathematical theory of the kinetics of the coagulation of colloidal solutions. *Zeitschrift für Physikalische Chemie*, 19:129–135, 1917.

[54] M. Stein and F. E. Kruis. Optimization of a transferred arc reactor for metal nanoparticle synthesis. *Journal of Nanoparticle Research*, 18:258, 2016.

[55] N. S. Tabrizi, M. Ullmann, V. A. Vons, U. Lafont, and A. Schmidt-Ott. Generation of nanoparticles by spark discharge. *Journal of Nanoparticle Research*, 11:315–332, 2009.

[56] P. Tandon and D. E. Rosner. Monte Carlo simulation of particle aggregation and simultaneous restructuring. *Journal of Colloid and Interface Science*, 213:273–286, 1999.

[57] S. Tsantilis and S. E. Pratsinis. Evolution of primary and aggregate particle-size distributions by coagulation and sintering. *AIChE Journal*, 46:407–415, 2000.

[58] M. Tschannen, O. Bachem, and M. Lucic. Recent advances in autoencoder-based representation learning. *arXiv preprint arXiv:1812.05069*, 2018.

[59] G. D. Ulrich. Theory of particle formation and growth in oxide synthesis flames. *Combustion Science and Technology*, 4:47–57, 1971.

[60] S. Vemury, K. A. Kusters, and S. E. Pratsinis. Time-lag for attainment of the self-preserving particle size distribution by coagulation. *Journal of Colloid and Interface Science*, 165:53–59, 1994.

[61] S. Vemury and S. E. Pratsinis. Self-preserving size distributions of agglomerates. *Journal of Aerosol Science*, 26:175–185, 1995.

[62] V. Vukotić, S.-L. Pintea, C. Raymond, G. Gravier, and J. C. van Gemert. One-step time-dependent future video frame prediction with a convolutional encoder-decoder neural network. In S. Battiato, G. Gallo, R. Schettini, and F. Stanco, editors, *Image Analysis and Processing*, pages 140–151. Springer, 2017.

[63] H. Wadell. Volume, shape, and roundness of quartz particles. *The Journal of Geology*, 43:250–280, 1935.

[64] G. G. Wang and S. Shan. Review of metamodeling techniques in support of engineering design optimization. In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, volume 4255, pages 415–426, 2006.

[65] K. Wang, S. Yu, and W. Peng. Extended log-normal method of moments for solving the population balance equation for Brownian coagulation. *Aerosol Science and Technology*, 53:332–343, 2019.

[66] J. Wei. A fast Monte Carlo method based on an acceptance-rejection scheme for particle coagulation. *Aerosol and Air Quality Research*, 13:1273–1281, 2013.

[67] D. L. Wright, R. McGraw, and D. E. Rosner. Bivariate extension of the quadrature method of moments for modeling simultaneous coagulation and sintering of particle populations. *Journal of Colloid and Interface Science*, 236:242–251, 2001.

[68] Y. Xiong and S. E. Pratsinis. Gas phase production of particles in reactive turbulent flows. *Journal of Aerosol Science*, 22:637–655, 1991.

[69] Y. Xiong and S. E. Pratsinis. Formation of agglomerate particles by coagulation and sintering—Part I. A two-dimensional solution of the population balance equation. *Journal of Aerosol Science*, 24:283–300, 1993.

[70] M. Zurita-Gotor and D. E. Rosner. Aggregate size distribution evolution for Brownian coagulation-sensitivity to an improved rate constant. *Journal of Colloid and Interface Science*, 274:502–514, 2004.

A Analytical solution for the Koch and Friedlander model

The reduction of surface area by sintering, defined in Eq. (9), is analytically solvable under certain conditions. For sufficiently small time steps $\Delta t = t_{i+1} - t_i$ we assume the characteristic sintering time τ_s to be constant, such that

$$\int_{a_{t_i}}^{a_{t_{i+1}}} \frac{1}{a - a_\mathrm{sph}} \mathrm{d}a = -\frac{1}{\tau_\mathrm{s}} \int_{t_i}^{t_{i+1}} \mathrm{d}t. \tag{38}$$

Solving the integrals yields

$$\ln \left(\frac{|a_{t_{i+1}} - a_\mathrm{sph}|}{|a_{t_i} - a_\mathrm{sph}|} \right) = -\frac{\Delta t}{\tau_\mathrm{s}}. \tag{39}$$

Here, $a_{t_{n+1}} = a_\mathrm{sph}$ and $a_{t_i} = a_\mathrm{sph}$ are introduced as exceptions. In those cases, the particle surface area is simply set to its theoretical minimum a_sph. Rearranging yields the solution

$$a_{t_{i+1}} (\Delta t) = a_\mathrm{sph} + (a_{t_i} - a_\mathrm{sph}) \exp \left(-\frac{\Delta t}{\tau_\mathrm{s}} \right). \tag{40}$$

Problems with this implementation could arise when applying to big temperature gradients as typically observed in flame or arc reactors since the analytical method could be incapable of yielding a continuous solution for these cases. Inaccuracies are also obtained at large time steps Δt, which is unlikely in settings with a lot of coagulation events as observed in the present work.

B Error propagation for χ

The ratio χ of characteristic coagulation and sintering time, defined in Eq. (15), is erroneous due to the deviations of τ_c and τ_s across all simulations N_Sims, which is computed through error propagation with

$$\delta\chi = \chi \sqrt{\left(\frac{\delta\tau_\mathrm{c}}{\tau_\mathrm{c}} \right)^2 + \left(\frac{\delta\tau_\mathrm{s}}{\tau_\mathrm{s}} \right)^2}. \tag{41}$$

Here $\delta\tau_\mathrm{s}$ is simply derived from twice the standard deviation between all simulations N_Sims. From the definition of the average characteristic coagulation time τ_c (Eq. (8)), we compute the error propagation for $\delta\tau_\mathrm{c}$ from the errors (twice the standard deviation) between all simulations N_Sims of the concentrations N_{t_i} and $N_{t_{i+1}}$, corresponding to the arbitrary time step $\Delta t = t_{i+1} - t_i$. Error propagation for τ_c yields

$$\delta\tau_\mathrm{c} = \sqrt{\left(\frac{\delta\tau_\mathrm{c}}{\delta N_{t_i}} \delta N_{t_i} \right)^2 + \left(\frac{\delta\tau_\mathrm{c}}{\delta N_{t_{i+1}}} \delta N_{t_{i+1}} \right)^2}. \tag{42}$$

The partial derivatives for $\delta\tau_\mathrm{c}/\delta N_{t_i}$ and $\delta\tau_\mathrm{c}/\delta N_{t_{i+1}}$ are defined as

$$\frac{\delta \tau_{\mathrm{c}}}{\delta N_{t_i}} = \frac{\delta}{\delta N_{t_i}} \left(\frac{N_{t_{i+1}}}{N_{t_i} - N_{t_{i+1}}} \Delta t \right) = -\frac{N_{t_{i+1}}}{\left(N_{t_i} - N_{t_{i+1}}\right)^2} \Delta t, \tag{43}$$

and

$$\frac{\delta \tau_{\mathrm{c}}}{\delta N_{t_{i+1}}} = \frac{\delta}{\delta N_{t_{i+1}}} \left(\frac{N_{t_{i+1}}}{N_{t_i} - N_{t_{i+1}}} \Delta t \right) = \frac{N_{t_i} - 2N_{t_{i+1}}}{\left(N_{t_i} - N_{t_{i+1}}\right)^2} \Delta t. \tag{44}$$

C Network architecture

This appendix provides an explicit definition of the neural network architecture of SAN. Table 3 presents a detailed overview of the number of layers and their respective parameters. In the following, each of these layers is formally defined. For simplicity, let $[K] = \{1, \ldots, K\}$ denote the set of positive integers less than or equal to K, and let X_i denote the value of a matrix X at position i.

The considered two-dimensional convolutional layer $\mathrm{Conv} \colon \mathbb{R}^{C_{\mathrm{in}} \times H \times W} \to \mathbb{R}^{C_{\mathrm{out}} \times \frac{H}{2} \times \frac{W}{2}}$, with kernel size 4 and stride 2, is defined for $(c, i, j) \in [C_{\mathrm{out}}] \times [\frac{H}{2}] \times [\frac{W}{2}]$ as

$$\mathrm{Conv}(x)_{c,i,j} = \sum_{c'=1}^{C_{\mathrm{in}}} \sum_{u=1}^{4} \sum_{v=1}^{4} w_{c,c',u,v} x_{c',2i+u,2j+v}. \tag{45}$$

This layer applies a learnable kernel $w \in \mathbb{R}^{C_{\mathrm{out}} \times C_{\mathrm{in}} \times 4 \times 4}$ to its input, detecting spatial patterns and reducing its spatial size $(H \mapsto \frac{H}{2}, W \mapsto \frac{W}{2})$.

In contrast, the transposed convolution $\mathrm{ConvTranspose} \colon \mathbb{R}^{C_{\mathrm{in}} \times H \times W} \to \mathbb{R}^{C_{\mathrm{out}} \times 2H \times 2W}$, with stride 2 and kernel size 4, expands the spatial size of its input $(H \mapsto 2H, W \mapsto 2W)$. This operation is sometimes referred to as inverse convolution, although it is not the mathematical inverse of convolution. For $(c, i, j) \in [C_{\mathrm{out}}] \times [2H] \times [2W]$, it is defined as

$$\mathrm{ConvTranspose}(x)_{c,i,j} = \sum_{c'=1}^{C_{\mathrm{in}}} \sum_{u=1}^{4} \sum_{v=1}^{4} w_{c,c',u,v} x_{c',\lfloor \frac{i-u}{2} \rfloor, \lfloor \frac{j-v}{2} \rfloor}, \tag{46}$$

where $\lfloor \cdot \rfloor : \mathbb{R} \to \mathbb{Z} = \{\ldots, -1, 0, 1, \ldots\}$ denotes rounding to the greatest integer less than or equal to the argument. The transposed convolution layer redistributes feature information onto a larger spatial grid, reconstructing structural details. Its trainable parameters are given by $w \in \mathbb{R}^{C_{\mathrm{out}} \times C_{\mathrm{in}} \times 4 \times 4}$.

The third type of layer with learnable parameters is the linear layer $\mathrm{Linear} \colon \mathbb{R}^m \to \mathbb{R}^n$, defined by

$$\mathrm{Linear}(x)_i = \sum_{j=1}^{m} W_{i,j} x_j, \tag{47}$$

for $i \in [n]$. Each output entry is thus a weighted sum of all input entries. The learnable parameters are given by $W \in \mathbb{R}^{n \times m}$.

To enable the network to represent non-linear functions, a non-linear activation layer is introduced. More precisely, the rectified linear unit activation layer $\mathrm{ReLU} \colon \mathbb{R}^{C \times H \times W} \to \mathbb{R}^{C \times H \times W}$ is defined as

$$\mathrm{ReLU}(x)_{c,i,j} = \max\left\{0, x_{c,i,j}\right\}, \tag{48}$$

for $(c, i, j) \in [C] \times [H] \times [W]$. This operation suppresses negative values while preserving positive activations.

To allow the network to process both an input image/histogram and a scalar-valued temperature, a concatenation layer is employed. For $x \in \mathbb{R}^m$ and $v \in \mathbb{R}$, it is given by

$$\mathrm{Concat}(x, v) = (x_1, x_2, \ldots, x_m, v). \tag{49}$$

Finally, to bridge the representations of linear layers (operating on flat vectors) and convolutional layers (operating on three-dimensional tensors), flattening and reshaping layers are introduced. They are formally defined as

$$\mathrm{Flatten}\colon \mathbb{R}^{C\times H\times W} \to \mathbb{R}^{CHW}, \quad \mathrm{Flatten}(x)_p = x_{c,i,j}, \tag{50}$$

where $(c, i, j) \in [C] \times [H] \times [W]$ is the unique solution of $p = (c-1)HW + (i-1)W + j$, and

$$\mathrm{Reshape}\colon \mathbb{R}^{CHW} \to \mathbb{R}^{C\times H\times W}, \quad \mathrm{Reshape}(x)_{c,i,j} = x_{(c-1)HW+(i-1)W+j}, \tag{51}$$

which is the inverse transformation of Flatten.

Table 3: Architecture of the Autoencoder Network

Layer Type	Input Dimension	Output Dimension	Number of Parameters
Conv	(1, 64, 64)	(32, 32, 32)	512
ReLU	(32, 32, 32)	(32, 32, 32)	0
Conv	(32, 32, 32)	(64, 16, 16)	32 768
ReLU	(64, 16, 16)	(64, 16, 16)	0
Conv	(64, 16, 16)	(128, 8, 8)	131 072
ReLU	(128, 8, 8)	(128, 8, 8)	0
Flatten	(128, 8, 8)	(8192)	0
Linear	(8192)	(128)	1 048 576
ReLU	(128)	(128)	0
Concat	(128) , (1)	(129)	0
Linear	(129)	(128)	16 512
Linear	(128)	(8192)	1 048 576
ConvTranspose	(128, 8, 8)	(64, 16, 16)	131 072
ReLU	(64, 16, 16)	(64, 16, 16)	0
ConvTranspose	(64, 16, 16)	(32, 32, 32)	32 768
ReLU	(32, 32, 32)	(32, 32, 32)	0
ConvTranspose	(32, 32, 32)	(1, 64, 64)	512