

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

185,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Operator Splitting Monte Carlo Method for Aerosol Dynamics

Kun Zhou and Tat Leung Chan

Additional information is available at the end of the chapter

<http://dx.doi.org/10.5772/65140>

Abstract

Aerosol dynamics are described by the population balance equation (PBE). In principle, three typical methods (i.e., direct discretization, method of moments, and stochastic method) have been widely used to solve the PBE. Stochastic method is the most flexible among the three methods. However, stochastic method is computationally expensive. Recently, an operator splitting Monte Carlo (OSMC) method has been developed so as to improve the numerical efficiency while preserving the flexibility of the stochastic method. Within the OSMC, nucleation and surface growth are handled with deterministic means, while coagulation is simulated with a stochastic method (the Marcus-Lushnikov stochastic process). The stochastic and deterministic treatments of various aerosol dynamic processes are synthesized under the framework of operator splitting. Here, the operator splitting errors of various schemes are analyzed rigorously, combined with concrete numerical examples. The analyses not only provide sound theoretical bases for selecting the most efficient operator splitting scheme for the usage of the OSMC, but also shed lights on how to adopt operator splitting in other PBE solving methods, i.e., direct discretization, method of moments, etc.

Keywords: Monte Carlo simulation, aerosol dynamics, operator splitting

1. Introduction

Aerosols are dispersed systems of solid or liquid particles suspended in air. Aerosols are created during various natural processes and industrial operations [1], such as clouds, smoke particles from forest fires, sand and dust storms, volcanic dust, spores and seeds from plant life, fluidized catalysts, spray drying of fluids, atomization of liquid fuels, fly ash from stacks, soot in combustion flames, etc. Aerosol evolution is very complex, involving various physical

and chemical processes. The most popular description of aerosol evolution is using the population balance equation (PBE) [2], which is also called the general dynamic equation in aerosol field [3]. The PBE is a convection-diffusion type equation with source terms taking into account the particle formation and depletion from various aerosol dynamics, mainly nucleation, surface growth (condensation), and coagulation.

Analytical solutions to the PBE are available for only a few specific cases [4–7]. Mostly, numerical means are resorted to solve the PBE. There are generally three typical methods for solving the PBE, i.e., direct discretization, method of moments, and stochastic methods. Direct discretization may be in the particle size space, such as discretization by sections [8] or finite difference/finite element [9] and may also be in the functional space, such as those in the collocation method [10], Galerkin method [11], and weighted residual method [12]. Among the various discretizations, the sectional method pioneered by Gelbard et al. [8] is the most popular. There are quite a few sectional methods developed aiming to alleviate the nonpositiveness and nonconservation problems found in earlier works [13], and to obtain better efficiency and flexibility, such as the fixed/moving pivot method [14, 15] and cell average method [16, 17].

Other than solving the discretized PBE, a widely used alternative approach is the method of moments, which solves a group of moments equations derived from the PBE. These moments (generally several low order moments) provide important information on the particle size distribution (PSD) function, i.e., number density, volume fraction, polydispersity, etc. However, owing to the nonlinearity of the PBE, the governing equations for lower order moments generally contain higher order moments, which are not closed after cutting off at a certain order of moment. Various closure methods have been developed, such as unimodal log-normal approximation [18], quadrature/direct quadrature [19, 20], interpolative closure [21], Taylor expansion [22], etc. Although the method of moments is computationally highly efficient, the complexity of the closure makes it difficult to accommodate complex physical models for aerosol dynamics with high flexibility. Another issue closely tied to method of moments is the realizability problem, when the moments sequence fails to find its corresponding distribution function [23, 24]. Among all the drawbacks, the inability to provide the PSD is the most severe, when the PSD is highly needed.

Stochastic simulation (or Monte Carlo simulation) does not solve the PBE, but to mimic the evolution of aerosol particles through a stochastic particle system. Based on the Marcus-Lushnikov process modeling of coagulation [29], various stochastic methods have been developed to simulate the general aerosol evolution. It is very difficult to categorize precisely so many diversified methods. According to the evolution of time, there is time-driven or event-driven Monte Carlo [30, 31]. According to the different ways to weight the numerical particles, there are the mass flow algorithms [32, 33], differentially weighted algorithm [34], and the more general weighted flow algorithms [35]. Also, there are various types of methods to improve the simulation efficiency, such as the majorant kernel method [36] and the τ leaping method [37].

Among the three typical methods, stochastic simulation is the most flexible for accommodating complex physical models for various aerosol dynamics, and is easy to implement without

special numerical issues, which are often encountered in the other two methods, e.g., nonpositiveness and nonconservation problems in direct discretization, realizability problem in method of moments. However, stochastic simulation generally has convergence rate proportional to $\sqrt{N}$, where N is the number of total numerical particles used. The simulation cost (CPU time and memory) is usually proportional to N^2 [25]. Hence, stochastic simulation is not numerically efficient when pursuing a high precision solution. Recently, Zhou et al. [26] developed an operator splitting Monte Carlo (OSMC) method, which combines stochastic simulation for coagulation process and deterministic integration for nucleation and surface growth (condensation). Stochastic simulation for coagulation is much more efficient than directly solving the Smoluchowski integro-differential equation (the coagulation model equation) with traditional discretization [27]. On the other hand, deterministic integration for nucleation and surface growth processes is far more efficient than stochastic simulations. The stochastic simulation and the deterministic integration in OSMC are synthesized under the framework of operator splitting. The accuracy and efficiency of the OSMC have been demonstrated through various testing cases [26]. Especially, the OSMC has been implemented with Message Passing Interface (MPI) to achieve very promising parallel simulation efficiency [26]. Here, the impact of various operator splitting schemes on the accuracy and efficiency of the OSMC is addressed.

2. Methodology

2.1. Governing equations

Aerosol particles are generally described statistically in a mesoscopic scale through defining the PSD, $n(\vec{\xi}, \vec{x}, t)$, and $n(\vec{\xi}, \vec{x}, t)d\xi$ denotes the number density of aerosol particles at location $\vec{x}$ at time t with aerosol properties between $\vec{\xi}$ and $\vec{\xi} + d\vec{\xi}$. Here, $\vec{\xi}$ is usually called an internal variable, which can be volume, surface, constituent, etc. For simplicity, it is assumed $\vec{\xi}$ to be the particle size (volume or diameter specified by the context). It is worth pointing out that the OSMC method can also deal with the cases with a vector type $\vec{\xi}$, i.e., combined properties to describe particles. In fact, the current OSMC code treats the diameter, surface, and volume of an aerosol particle independently, which can easily model particles of various morphologies, such as particles with fractal dimensions. The evolution of the PSD function satisfies the PBE (also called the general dynamic equation) [3]

$$\frac{\partial n}{\partial t} + \nabla \cdot n\vec{u} = \nabla \cdot D\nabla n + I_{\text{nuc}} + G_{\text{cond}} + C_{\text{coag}}, \quad (1)$$

where $\vec{u}$ is the gas flow velocity, and D is the diffusion coefficient. The source terms of I_{nuc} , G_{cond} , and C_{coag} on the right hand side denote nucleation, surface growth (condensation), and coagulation, respectively. The determination of the convection velocity $\vec{u}$ requires solving the

fluid dynamics control equations, and the coupling of the fluid dynamics with the aerosol dynamics is another important research field. Here, the gas velocity $\vec{u}$ is assumed known, using some techniques (often computational fluid dynamics). It is worth pointing out that the coupling between the fluid dynamics and the aerosol dynamics is generally handled by the operator splitting method [11, 28–30], i.e., to solve the fluid dynamics and the aerosol dynamics one by one with the feedback from the other dynamics temporarily freezing, and to iterate between these two dynamics to obtain a converged solution.

Another simplification in this work is to neglect the diffusion term in the PBE. For aerosol particles, the mass diffusion due to the number concentration gradient ($\nabla \cdot D \nabla n$) is usually very small compared to the air viscous diffusion, which can be described by the Schmidt number (i.e., the ratio of the viscous diffusion to the mass diffusion). For example, under standard atmospheric conditions, a spherical particle with a diameter of 10 nm has a Schmidt number equal to 290, and a 100 nm particle has Schmidt number 2.2×10^4 [3]. So it is legitimate to neglect the aerosol mass diffusion under very general conditions.

With the assumptions that the gas velocity $\vec{u}$ is known and the mass diffusion term is negligible, the PBE can be simplified as

$$\frac{\partial n}{\partial t^*} = I_{\text{nuc}} + G_{\text{cond}} + C_{\text{coag}}. \quad (2)$$

The convection term on the left hand side of the original PBE (1) has been implicitly solved with the introduction of the Lagrangian time t^*

$$t^*(\vec{x}) = \int_0^{\vec{x}} \frac{d\vec{x}'}{\vec{u}(\vec{x}')}, \quad (3)$$

The transformation converts the Eulerian point of view in Eq. (1) to the Lagrangian point of view in Eq. (2), with the help of the assumed known velocity field $\vec{u}$. There are many practically interested conditions, where the flow field can be considered as steady and one-dimensional, e.g., burner stabilized flames and counter flow diffusion flames. Then it is very easy to integrate Eq. (3) to build up a one-to-one mapping between the time and the spatial coordinate. Hence, the convection can be readily assimilated in the model Eq. (2). For general cases of aerosol evolution in three dimensional flows, it is possible to use the Lagrangian particle scheme to simulate the aerosol evolution along Lagrangian trajectories [31, 32]. So the model Eq. (2) and the subsequent OSMC method may have very broad applications. Later on, t^* is simply denoted as t , since the Lagrangian point of view is not explicitly referred any longer.

The aerosol dynamic processes on the right hand side describe the interactions among molecules (or cluster of molecules) and particles. Nucleation is the process that dozens or hundreds of molecules form a stable critical nucleus. Surface growth includes physical

condensation and surface chemical reactions, which involves interactions between gas phase molecules and an aerosol particle. Surface chemical reactions will not be discussed here due to their vastness. Coagulation is the process that two particles coalesce to form a new one. More details about the aerosol dynamic processes are presented in Section 2.2.

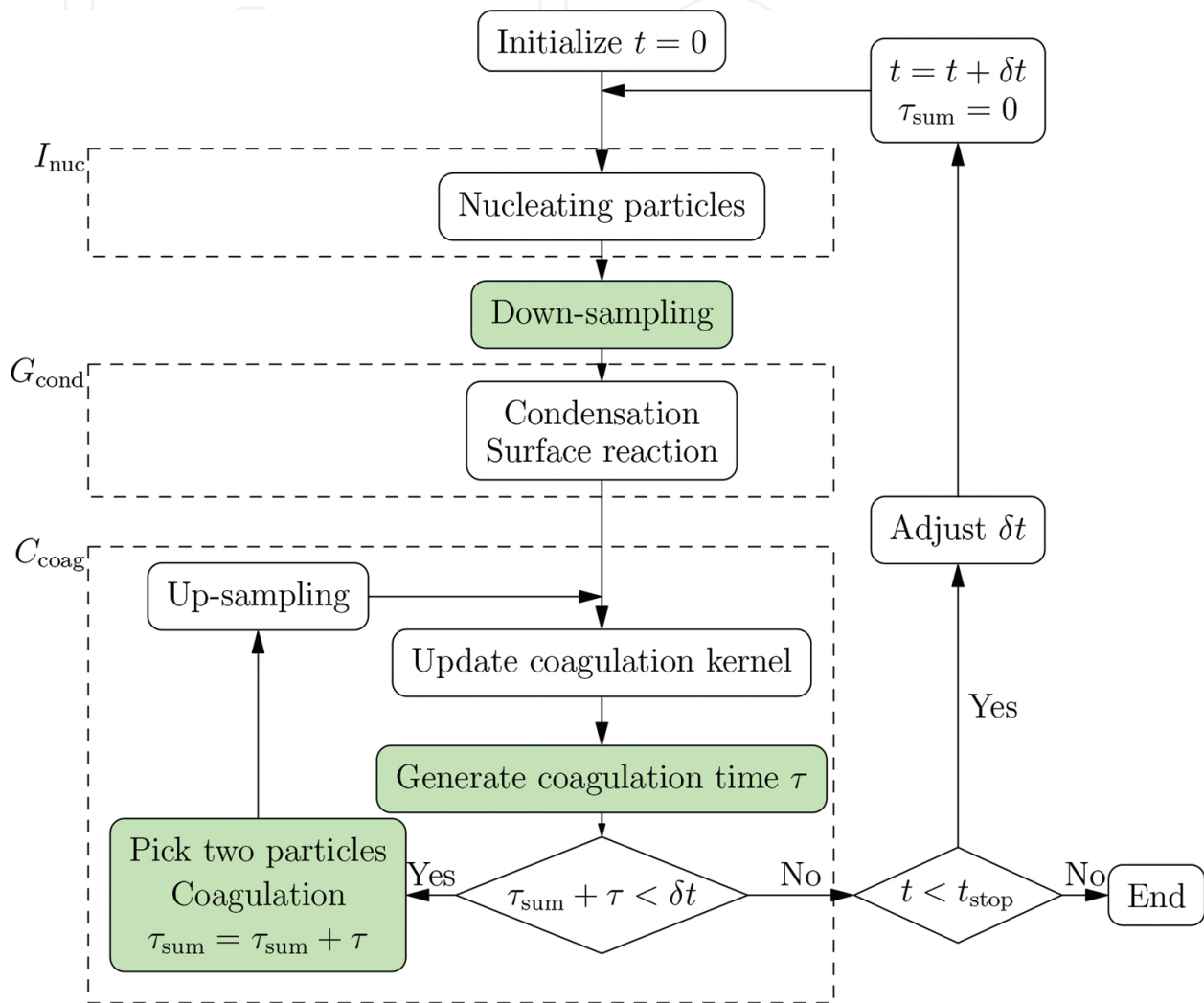


Figure 1. Flowchart of the OSMC. Filled blocks denote steps where stochasticity arises. Adapted from Reference [26].

2.2. Operator splitting Monte Carlo method

Operator splitting is an efficient and powerful method to solve such evolution equations [33] as Eq. (2). In an operator splitting method, the time evolution of $n(t)$ is still handled by the classical methods, such as the forward Euler method or Runge-Kutta method. Other than to integrate all aerosol dynamic processes together in one time step δt , operator splitting separates the integration into multiple fractional steps. There are various ways to construct the multiple fractional steps, such as $I_{\text{nuc}} \rightarrow G_{\text{cond}} \rightarrow C_{\text{coag}}$ or $\frac{1}{2}I_{\text{nuc}} \rightarrow G_{\text{cond}} \rightarrow \frac{1}{2}C_{\text{coag}}$, where $1/2$ denotes a

half integration step. The flowchart (**Figure 1**) shows an example how the operator splitting scheme $I_{\text{nuc}} \rightarrow G_{\text{cond}} \rightarrow C_{\text{coag}}$ is carried out. Details of the various models for the aerosol dynamics are described in the following subsections, and the error analyses for various operator splitting schemes are given in Section 3.

2.2.1. Nucleation

Nucleation is the gas-to-particle transition to generate nascent nanoparticle. There are various nucleation models, largely based on the classical nucleation theory developed in the 1920–1940s [34]. In the classical nucleation theory, there are two key parameters for the nucleation modeling, i.e., the nucleation rate (how many particles are nucleated per unit volume, unit time), and the critical size, which is the minimum size of nucleated particles with stable nucleus. According to the classical Becker-Döring theory [3, 35], the nucleation rate I and critical diameter d^* are

$$I = \frac{P_v x_v}{k_B T} \sqrt{\frac{2\sigma}{\pi m}} \exp\left(-\frac{16\pi\sigma^3 m^2}{3(k_B T)^3 \rho_p^2 (\ln S)^2}\right), \quad (4)$$

$$d_p^* = \frac{4\sigma v_m}{k_B T \ln S}. \quad (5)$$

Here, P_v is the vapor pressure, x_v is the mole fraction of the vapor, m is the molecular mass of the nucleation vapor, ρ_p is the particle density, T is the temperature, $S = P_v/P_{\text{sat}}$ is the vapor saturation ratio, k_B is the Boltzmann constant, σ is the surface tension, and $v_m = m/\rho_p$ is the molecular effective volume of the nucleation vapor. Hence, the nucleation term in Eq. (2) can be expressed with the help of the Dirac delta function as

$$I_{\text{nuc}} = \int I \delta(d_p - d_p^*) dd_p, \quad (6)$$

which means that only particles with the critical size are nucleated.

Within the OSMC, nucleation model is not limited to the classical nucleation theory. Any other nucleation model can be readily included, should only the nucleation rate and the critical size be provided.

2.2.2. Surface growth (condensation)

The functional form of the particle volume growth rate due to condensation depends on the Knudsen number $\text{Kn} = 2\lambda/d_p$, i.e., the ratio between the air mean free path λ and the particle

radius $d_p/2$. In the free molecular regime ($\text{Kn} \geq 10$) and the continuum regime ($\text{Kn} \leq 0.1$), the condensational growth rates can be derived from the gas kinetic theory and the gradient diffusion model, respectively. In the transition regime ($0.1 < \text{Kn} < 10$), interpolation formulas are usually used, such as the harmonic mean of the free molecular and continuum regimes. The expressions for volume growth rate G_v and the condensation term G_{cond} are [3]

$$G_v = \begin{cases} \frac{\pi d_p^2 (P_v - P_p) v_m}{(2\pi m k_B T)^{1/2}}, & \text{Kn} \geq 10 \\ \frac{2\pi D d_p (P_v - P_p) v_m}{k_B T}, & \text{Kn} \leq 0.1 \\ \text{harmonic mean}, & 0.1 < \text{Kn} < 10 \end{cases} \quad (7)$$

$$G_{\text{cond}} = -\frac{\partial(nG_v)}{\partial v}. \quad (8)$$

Here, P_p is the equilibrium vapor pressure for particles of size d_p , which is usually assumed to be the saturation pressure of the condensation vapor. For particles close to the critical size, the Kelvin correction is also required [3].

Within the OSMC, any other surface growth model can also be accommodated, should only the particle volume growth rate be provided.

2.2.3. Stochastic simulation of coagulation

Coagulation is the process that two particles coalesce/aggregate to form a new particle. The coagulation dynamics is modeled by the well-known Smoluchowski's equation [3]

$$C_{\text{coag}} = \frac{1}{2} \int_0^v \beta(v, v - \tilde{v}) n(\tilde{v}) n(v - \tilde{v}) d\tilde{v} - \int_0^\infty \beta(v, \tilde{v}) n(v) n(\tilde{v}) d\tilde{v}. \quad (9)$$

The coagulation kernel function $\beta(v, u)$ describes the rate at which particles of size v coagulate with particles of size u . The first term on the right-hand side denotes the formation rate of particles with volume v due to coagulation between particles $\tilde{v}$ and $v - \tilde{v}$. (Volume is conserved before and after coagulation.) The half factor ($1/2$) is to correct the double-counting effect. The second term on the right-hand side describes the depletion rate of particles with volume v due to their coagulation with any other particles. Practically, the coagulation kernel function $\beta(v, u)$ is very hard to find, except for simplified conditions. In the free molecule regime ($\text{Kn} \geq 10$), β has the form [3]

$$\beta(v,u) = \left(\frac{6}{\pi}\right)^{2/3} \left(\frac{\pi k_B T}{2\rho_p}\right)^{1/2} \left(\frac{1}{v} + \frac{1}{u}\right)^{1/2} (v^{1/3} + u^{1/3})^2. \quad (10)$$

In the continuum regime ($\text{Kn} \leq 0.1$), β takes the form [3]

$$\beta(v,u) = \frac{2k_B T}{3\mu} \left(\frac{1}{v^{1/3}} + \frac{1}{u^{1/3}}\right) (v^{1/3} + u^{1/3}), \quad (11)$$

where μ is the dynamic viscosity of the gas-phase. In the transition regime ($0.1 < \text{Kn} < 10$), interpolation formula taking into account the former two regimes are generally used.

Although the Smoluchowski's equation was developed one century ago, and it has received broad attention in diverse fields, no analytical solutions are known for the kernels discussed above. Analytical solutions known so far are only limited to the following three simple kernels (or their linear combination) [5], i.e., (i) $\beta(v,u) = 1$, (ii) $\beta(v,u) = v + u$, and (iii) $\beta(v,u) = vu$.

Other than to solve the Smoluchowski's equation directly through various numerical means, the coagulation process is simulated by the Marcus-Lushnikov stochastic process in the OSMC. The Marcus-Lushnikov process is the natural stochastic analogue of the finite discrete Smoluchowski coagulation equation. The well-known review paper [5] provides a wide-ranging survey on the Marcus-Lushnikov process and the Smoluchowski's equation. Here, we follow the work of Gillespie [25], and give an intuitive introduction to the stochastic simulation of coagulation.

In a virtual box of volume V , there are N "well-mixed" particles. Any two particles may coalesce/aggregate to form a new one at a random time. The propensity C_{ij} of coalescence between two particles i and j is determined by the coagulation kernel $C_{ij} = \beta(i,j)/V$. Let $P(\tau, i, j) d\tau$ denote the probability that the next coalescence will occur in the time interval $(\tau, \tau + d\tau)$ between i and j ($i < j$), then it is found [25]

$$P(\tau, i, j) = C_{ij} \exp \left[- \sum_{k=1}^{N-1} \sum_{l=k+1}^N C_{kl} \tau \right]. \quad (12)$$

In essence, the simulation is to randomly pick up two particles i and j to coalesce (forming a new particle) at the random time τ according to the joint probability density function $P(\tau, i, j)$. Repeat the former process as time advances. During the simulation, the number of particles drops, and the particle number density drops also.

The key step in the simulation is to generate a random point (τ, i, j) according to the density function $P(\tau, i, j)$. The random coagulation time τ satisfies a Poisson process (the exponential distribution), and it can be generated as

$$\tau = \frac{1}{C_0} \ln \left(\frac{1}{r_1} \right), \quad (13)$$

where $C_0 = \sum_{i=1}^{N-1} \sum_{j=i+1}^N C_{ij}$, and r_1 is a random number picked from the uniform distribution in the unit interval. Eq. (13) gives the classical way to generate a random variate from the exponential distribution with parameter C_0 .

After selecting the time at which two particles coagulate, the next step is to randomly select two particles according to the probability function $P_2(i, j) = C_{ij}/C_0$. Loosely speaking, C_0 is the total coagulation probability between any two particles, and $P_2(i, j)$ is the fraction of the coagulation probability C_{ij} over C_0 . Gillespie [25] introduced two methods to choose i and j , the so-called full conditioning and partial conditioning methods. In the full conditioning method, the selection of i and j is accomplished in two steps. During the first step, calculate the partial sum $C_i = \sum_{k=i+1}^N C_{ik}$ ($i = 1, \dots, N$), then combine the N partial sums in sequential to form a range with the index i from 1 to N denoting the part for each partial sum, and then it is straightforward to build up a linear mapping between the range and the unit interval (0,1). Draw a random number from the uniform distribution, and check the random number landing on which corresponding part in the range, and then select particle i by the part index. During the second step, combine C_{ij} , ($j = i + 1, \dots, N$) in sequential to form another range, and build up the linear mapping between the new range with the unit interval, then draw another random number to determine the particle j in a similar way as in the first step. In the partial conditioning method, it is actually an acceptance-rejection method to generate a bivariate random number (i, j) from the function $P_2(i, j)$. In a general acceptance-rejection algorithm to generate a vector random variable $\vec{x}$ with density $f(\vec{x})$, it needs to choose a density function $g(\vec{x})$ and a constant $c \geq 1$ such that $cg(\vec{x}) \geq f(\vec{x})$. The algorithm proceeds as follows [36]:

1. Generate $\vec{Y}$ from density g .
2. Generate $\vec{U}$ from the uniform distribution on the interval $(0, cg(\vec{Y}))$.
3. If $\vec{U} \leq f(\vec{Y})$, then set $\vec{X} = \vec{Y}$; else, return to 1.

The density function $g(\vec{x})$ can be chosen freely. However, the optimal choice is to make the interval in 2 as small as possible, so that there is a better chance that $\vec{U} \leq f(\vec{Y})$ so as to accept $\vec{Y}$ as the required $\vec{X}$ without too often rejections and repetitions of the whole process. In the acceptance-rejection method [25] to generate i and j according to $P_2(i, j)$, $g(\vec{x})$ is chosen as a uniform function, and c is an upper bound of all $C_{i,j}$ (preferably the smallest upper bound). It is argued [25] that when the values of C_{ij} are relatively comparable with each other (ideally

constant) the particle conditioning method should be more efficient than the full conditioning method.

Within the OSMC, both the full and particle conditioning methods have been implemented to choose from easily.

2.3. Full algorithm for the OSMC

Figure 1 shows the flowchart of OSMC including all aerosol dynamics. Only the first-order operator splitting is sketched in the figure.

At time $t = 0$, the simulator is initialized with given parameters, i.e., number of Monte Carlo (MC) particles N , simulator size V , and the initial PSD. The simulator size is determined as $V = N/M_0$, where M_0 is the initial number density. If simulation starts from an empty case, i.e., $M_0 = 0$, then the simulator is initialized with N particles of uniform size (the real value of the size is immaterial to the simulation results), and the simulator size is set to a huge value, say $V = 10^{10}$, which renders a tiny particle number density to approximate the condition $M_0 = 0$. If simulation starts from a case with a specific PSD, then the size of the N simulation particles is assigned by a stochastic process to satisfy the initial PSD. There are various convenient ways to generate random number to satisfy a given distribution [37], although they are usually not necessary in Monte Carlo simulation of aerosol dynamics, since mostly the simulation starts from an empty case.

In the particles nucleation step, the simulator size V is adjusted to reflect the change of particle number density due to nucleation. Nascent nucleated particles are added to the pool of MC particles. If the total number of MC particles exceeds the maximum allowable threshold value (which is set beforehand to balance the stochastic error and numerical cost), then down-sampling is performed, i.e., exceeding MC particles are randomly removed from the pool to satisfy the number constraint. And then every MC particle undergoes the surface growth process according to its growth rate (usually depends on its size).

The coagulation simulation process (those steps grouped in the dashed bounding box X_s in **Figure 1**) shows how to implement the stochastic algorithm of Gillespie [25]. Updating coagulation kernel is to calculate the pair coagulation rate $C_{ij} = \beta(v_i, v_j)/V$, ($i = 1, \dots, N-1$, $j = i+1, \dots, N$). The random coagulation time τ is generated according to Eq. (13). The comparison statement $\tau_{\text{sum}} + \tau < \delta t$ is to judge whether the time for two particles to coagulate is still admissible within the discrete time step δt . Here τ_{sum} is the accumulated coagulation time in the coagulation step, which is initialized to zero before a coagulation step begins. If the comparison statement is true, two particles are selected to perform coagulation, and the number of MC particles decreases by one. The subsequent up-sampling step is to keep the number of MC particles above a given threshold to avoid severe stochastic error.

3. Error analyses for operator splitting

Aerosol dynamics usually contain nucleation, surface growth, and coagulation processes. These processes have very different effects on the evolution of the total number density and volume fraction, which are the two most important statistics of aerosol particles. For the number density, it increases due to nucleation while decreases due to coagulation, and it does not change in the surface growth process (mainly condensation). For the volume fraction, it is conserved in the coagulation process. The volume fraction is generally dominated by surface growth process, and nucleation has limited contribution to the volume fraction directly, because nucleated particles are very small in size.

Nucleation is generally a function of the temperature and the nucleated vapor concentration, and it does not depend on the aerosol status (number density, particle size, etc). However, both condensation and coagulation are closely related to the number density, which is largely determined by the nucleation process. On the other hand, condensation and coagulation are usually loosely coupled through particle size and number density. Therefore, it is convenient and meaningful to consider two isolated cases: (i) nucleation + coagulation, which determine the number density, and (ii) nucleation + condensation, which determine the volume fraction.

3.1. Case of nucleation + coagulation

Before discussing the case of nucleation and coagulation, we introduce an integral transformation of the Smoluchowski's equation (9), which is very useful to understand the evolution of the moments of the PSD. Suppose $F(v)$ is an arbitrary function of v , and let

$$M(t) = \int_0^\infty F(v)n(v,t)dv, \quad (14)$$

then from the Smoluchowski's equation (9), it can be derived [1]

$$\frac{dM(t)}{dt} = \frac{1}{2} \int_0^\infty \int_0^\infty [F(v+u) - F(v) - F(u)]\beta(v,u)n(v,t)n(u,t)dudv. \quad (15)$$

Specifically, when $F(v) = v^k$, ($k \geq 0$), $M_k(t)$ is the k th order moment, and Eq. (15) transforms to the dynamic equation for the moment $M_k(t)$

$$\frac{dM_k(t)}{dt} = \frac{1}{2} \int_0^\infty \int_0^\infty [(v+u)^k - v^k - u^k]\beta(v,u)n(v,t)n(u,t)dudv. \quad (16)$$

One interesting but somewhat overlooked conclusion [1] is that the right hand side of Eq. (16) is negative if $k < 1$, and it is positive if $k > 1$, which mean that M_k will grow with time for

$k > 1$, and it will decrease with time for $k < 1$. Especially, $M_1(t)$ is constant with time. Actually, M_1 is the volume fraction, which is conserved during the pure coagulation process. M_0 is the total number density, which does decay due to coagulation. It is worth pointing out that although fractional moments (with k being a fraction number) do not possess obvious physical meaning, they are very useful in the method of moments with the interpolative closure [21] or with the Taylor expansion closure [22].

In the case of constant rate nucleation and constant kernel coagulation, the number density (M_0) satisfies the following simple model equation

$$\frac{dM_0}{dt} = J + KM_0^2. \quad (17)$$

On the right-hand side, the first term models the contribution to the number density from nucleation, and J represents the nucleation rate (number of particles nucleated per unit volume per unit time). The second term models the contribution from coagulation, and $K = -\frac{1}{2}\beta_0$, where β_0 is a constant coagulation kernel. In fact, in the moment dynamic Eq. (16) if the coagulation kernel is the constant β_0 and let $k = 0$ (to obtain the number density equation); we immediately have $-\frac{1}{2}\beta_0 M_0^2$ on the right-hand side. The notation $K = -\frac{1}{2}\beta_0$ (be aware of the negative sign) is found to be able to greatly simplify the following analyses.

The assumption of a constant nucleation rate and a constant coagulation kernel does not severely limit the applicability of the following discussion on the operator splitting errors for the more general case of nucleation and coagulation. The analyses focus on the local error in a generally small single time step. Hence, even a time-dependent nucleation rate $J(t)$ can be readily assumed constant within the single time step. On the other hand, the coagulation kernel β is a bivariate function of colliding particles sizes. Mathematically, it is possible to find a β_0 that satisfies

$$\beta_0 M_0^2 = \beta_0 \int_0^\infty \int_0^\infty n(v,t)n(u,t)dudv = \int_0^\infty \int_0^\infty \beta(v,u)n(v,t)n(u,t)dudv. \quad (18)$$

Of course, the right β_0 depends not only on the specific β function, but also on the distribution n . Here, we assume that a good estimate of β_0 is available and will not delve into this issue further.

Given the initial condition $M_0(t = 0) = M_0^*$, the model Eq. (17) has analytical solution

$$M_0(t) = \frac{\sqrt{JK}}{K} \tan \left[t\sqrt{JK} + \arctan \left(\frac{M_0^* K}{\sqrt{JK}} \right) \right]. \quad (19)$$

In fact, a solution of the same Eq. (17) is found as [7]

$$M_0(t) = \frac{\sqrt{-JK}}{K} \tanh \left[t\sqrt{-JK} + \operatorname{arctanh} \left(\frac{M_0^* K}{\sqrt{-JK}} \right) \right]. \quad (20)$$

Since, $K = -\frac{1}{2}\beta_0 < 0$ the solution Eq. (19) involves imaginary numbers. Using their relation to the complex exponential function, it is easy to show that Eqs. (19) and (20) are actually equivalent. Although the solution Eq. (20) (with K replaced by $-\frac{1}{2}\beta_0$) is more natural, the form in Eq. (19) proves more handy in the following analyses. Here, we expand Eq. (19) at $t = \delta_t$ as Taylor series for later usage

$$M_0(\delta_t) = M_0^* + (KM_0^{*2} + J)\delta_t + K(KM_0^{*2} + J)M_0^*\delta_t^2 + \frac{1}{3}K(3K^2M_0^{*4} + 4JKM_0^{*2} + J^2)\delta_t^3 + \mathcal{O}(\delta_t^4). \quad (21)$$

On the other hand, we can try to use the operator splitting method to solve the same Eq. (17), and compare the obtained solutions with the true solution (19) to quantify the operator splitting errors. There are various operator splitting schemes. Here, we will focus on two first-order Lie schemes and two second-order Strang schemes, i.e., (i) $I_{\text{nuc}} \rightarrow C_{\text{coag}}$ (ii) $C_{\text{coag}} \rightarrow I_{\text{nuc}}$ (iii) $\frac{1}{2}I_{\text{nuc}} \rightarrow G_{\text{coag}} \rightarrow \frac{1}{2}I_{\text{nuc}}$ and (iv) $\frac{1}{2}C_{\text{coag}} \rightarrow I_{\text{nuc}} \rightarrow \frac{1}{2}C_{\text{coag}}$.

In the operator splitting method, the following two joint equations are solved

$$\frac{dM_0^{(1)}}{dt} = J, \quad (22a)$$

$$\frac{dM_0^{(2)}}{dt} = KM_0^2. \quad (22b)$$

(i) Lie scheme $I_{\text{nuc}} \rightarrow C_{\text{coag}}$ (Denoted as Lie1)

To seek the solution at $t = \delta_t$, Eq. (22a) is solved first with the given initial condition at $t = 0$, then Eq. (22b) is solved with the just obtained solution at $t = \delta_t$ as the initial condition.

Eq. (22a) has solution:

$$M_0^{(1)}(t) = Jt + M_0^*. \quad (23)$$

Hence, $M_0^{(1)}(\delta_t) = J\delta_t + M_0^*$. Take $M_0^{(1)}(\delta_t)$ as the initial condition and solve Eq. (22b), then the final solution is found as

$$M_0^{(\text{OP})}(t) = -\frac{J\delta_t + M_0^*}{JK\delta_t + KM_0^*t - 1}. \quad (24)$$

Let $t = \delta_t$ in the above equation, and expand it into Taylor series

$$M_0^{(\text{OP})}(\delta_t) = M_0^* + (KM_0^2 + J)\delta_t + K(KM_0^{*2} + 2J)M_0^*\delta_t^2 + \mathcal{O}(\delta_t^3). \quad (25)$$

The operator splitting error can be found by comparing the two Taylor expansion series (25) and (21)

$$Er_{0i}) = M_0^{(\text{OP})}(\delta_t) - M_0(\delta_t) = M_0^*JK\delta_t^2 + \mathcal{O}(\delta_t^3) = -\frac{1}{2}M_0^*J\beta_0\delta_t^2 + \mathcal{O}(\delta_t^3). \quad (26)$$

(ii) Lie scheme $I_{\text{coag}} \rightarrow C_{\text{nuc}}$ (Denoted as Lie2)

The analysis is quite similar to case (i), except that the solution procedure for the two joint Eqs. (22a) and (22b) is reversed. The final solution is

$$M_0^{(\text{OP})}(\delta_t) = J\delta_t - \frac{M_0^*}{KM_0^*\delta_t - 1}, \quad (27)$$

and its Taylor expansion is

$$M_0^{(\text{OP})}(\delta_t) = M_0^* + (KM_0^2 + J)\delta_t + K^2M_0^{*3}\delta_t^2 + \mathcal{O}(\delta_t^3). \quad (28)$$

Hence, the operator splitting error is

$$Er_{0ii}) = -M_0^*JK\delta_t^2 + \mathcal{O}(\delta_t^3) = \frac{1}{2}M_0^*J\beta_0\delta_t^2 + \mathcal{O}(\delta_t^3). \quad (29)$$

(iii) Strang scheme $\frac{1}{2}I_{\text{nuc}} \rightarrow C_{\text{coag}} \rightarrow \frac{1}{2}I_{\text{nuc}}$ (Denoted as Strang1)

Here, the solution to the joint Eqs. (22) is carried out in three steps. First, Eq. (22a) is solved with the initial condition $M_0(0) = M_0^*$ to get the solution $M_0^{(1)}(t)$. Second, $M_0^{(1)}(\frac{1}{2}\delta_t)$ is used as the initial condition to solve Eq. (22b) to get the solution $M_0^{(2)}(t)$. Finally, $M_0^{(2)}(\delta_t)$ is used as the initial condition at $t = \frac{1}{2}\delta_t$ to solve Eq. (22a) again. The final solution at $t = \delta_t$ is found as

$$M_0^{(\text{OP})}(\delta_t) = \frac{1}{2}J\delta_t - \frac{J\delta_t + 2M_0^*}{JK\delta_t^2 + 2KM_0^*\delta_t - 2}, \quad (30)$$

and its Taylor expansion is

$$M_0^{(\text{OP})}(\delta_t) = M_0^* + (KM_0^{*2} + J)\delta_t + K(KM_0^{*2} + J)M_0^*\delta_t^2 + \frac{1}{4}K(4K^2M_0^{*4} + 6KJM_0^{*2} + J^2)\delta_t^3 + \mathcal{O}(\delta_t^4). \quad (31)$$

Hence, the operator splitting error is

$$Er_{0\text{iii}} = \frac{JK}{12}(2M_0^{*2}K - J)\delta_t^3 + \mathcal{O}(\delta_t^4) = \frac{J\beta_0}{24}(M_0^{*2}\beta_0 + J)\delta_t^3 + \mathcal{O}(\delta_t^4). \quad (32)$$

(iv) Strang scheme $\frac{1}{2}C_{\text{coag}} \rightarrow I_{\text{nuc}} \rightarrow \frac{1}{2}C_{\text{coag}}$ (Denoted as Strang2)

Similar to case (iii), the final solution is found as

$$M_0^{(\text{OP})}(\delta_t) = 2 \frac{JKM_0^*\delta_t^2 - 2J\delta_t - 2M_0^*}{-JK^2M_0^*\delta_t^3 + 2JK\delta_t^2 + 4KM_0^*\delta_t - 4}, \quad (33)$$

and its Taylor expansion is

$$M_0^{(\text{OP})}(\delta_t) = M_0^* + (KM_0^{*2} + J)\delta_t + K(KM_0^{*2} + J)M_0^*\delta_t^2 + \frac{1}{4}K(4K^2M_0^{*4} + 5KJM_0^{*2} + 2J^2)\delta_t^3 + \mathcal{O}(\delta_t^4). \quad (34)$$

Hence, the operator splitting error is

$$Er_{0\text{iv}} = \frac{JK}{12}(-M_0^{*2}K + 2J)\delta_t^3 + \mathcal{O}(\delta_t^4) = -\frac{J\beta_0}{48}(M_0^{*2}\beta_0 + 4J)\delta_t^3 + \mathcal{O}(\delta_t^4). \quad (35)$$

To summarize, the two Lie schemes have the same magnitude in splitting error, both are of second order in δ_t within a time step. So the global integration error during time t_{all} is of order δ_t , since the error will accumulate in t_{all}/δ_t steps. The Lie1 scheme underestimates the number density, while the Lie2 scheme overestimates that. This is because coagulation rate closely depends on the number density, the higher the number density, the faster the coagulation rate. Coagulation rate would be overpredicted if nucleation is integrated before coagulation, hence the number density would be underpredicted as in Lie1, and vice versa.

The splitting errors in the two Strang schemes are of third order in δ_t within a time step, which renders them a second-order scheme globally. The Strang1 scheme is found to overestimate the number density, while the two Strang2 scheme to underestimate that. The magnitude of the absolute errors in these two schemes can be compared through investigating the coefficients of δ_t^3 in Eqs. (32) and (35)

$$2(M_0^{*2}\beta_0 + J) - (M_0^{*2}\beta_0 + 4J) = M_0^{*2}\beta_0 - 2J. \quad (36)$$

If $\frac{1}{2}M_0^{*2}\beta_0 < J$, i.e., coagulation rate is smaller than nucleation rate, the absolute splitting error in Eq. (32) is smaller than that in Eq. (35). When $\frac{1}{2}M_0^{*2}\beta_0 \ll J$, the error in Eq. (33) approaches a half of that in Eq. (36). On the other hand, when $\frac{1}{2}M_0^{*2}\beta_0 \gg J$, the error in Eq. (32) nearly doubles that in Eq. (35). Hence, in the Strang schemes it is preferable to split the higher rate process into symmetric substeps to reduce the splitting error.

Only the most common first and second-order operator splitting schemes are discussed so far. They are systematical ways to construct even higher order schemes. Especially, a third-order scheme [38] can be obtained through the combination of the two Lie and two Strang schemes. Here, we are not interested to discuss the even higher order schemes other than the Lie and Strang schemes, since higher order schemes are complicate to implement and may not be worthy for the solving the aerosol dynamics. One interesting conclusion [38] is that the scheme $\frac{2}{3}(\text{Strang1} + \text{Strang2}) + \frac{1}{4}(\text{Lie1} + \text{Lie2})$ is of third order, which can be used to check the accuracy of the above derivations. Keep the third-order term δ_t^3 in the error functions (26), (29), (32), and (29), then calculate the error in the third-order scheme constructed as mentioned shortly before, it is found that only terms involving δ_t^4 and higher orders survive. This is an indirect proof that all the derivations of the error functions are correct.

Following, we will briefly discuss the splitting errors for higher order moments other than M_0 . Since the total volume is conserved during coagulation, nucleation, and coagulation are independent of each other with respect to M_1 (volume fraction). Hence, there are no splitting errors for predicting M_1 for M_2 , the model equation for nucleation and coagulation can be easily derived from Eqs. (6) and (16) as

$$\frac{dM_2}{dt} = Jv^{*2} + \frac{1}{2}\beta_0 \int_0^\infty \int_0^\infty (2vu)n(v,t)n(u,t)dudv = Jv^{*2} + M_1^2, \quad (37)$$

and M_1 is determined by nucleation only, which can be integrated as

$$M_1(t) = Jv^*t + M_1^*. \quad (38)$$

Here, v^* is the critical volume of nucleated particle, and M_1^* is the initial value at $t = 0$. The solution to Eq. (37) is

$$M_2(t) = \frac{1}{3}J^2v^{*2}\beta_0t^3 + Jv^*M_1^*\beta_0t^2 + (Jv^{*2} + \beta_0M_1^{*2})t + M_2^*. \quad (39)$$

Similar to the above error analyses, the four splitting errors for solving Eq. (37) can be readily obtained as

$$Er_{2(i)} = \frac{1}{3}\beta_0Jv^*(2Jv^*\delta_t + 3M_1^*)\delta_t^2, \quad (40a)$$

$$Er_{2(ii)} = -\frac{1}{3}\beta_0Jv^*(Jv^*\delta_t + 3M_1^*)\delta_t^2, \quad (40b)$$

$$Er_{2(iii)} = -\frac{1}{12}\beta_0J^2v^{*2}\delta_t^3, \quad (40c)$$

$$Er_{2(iv)} = \frac{1}{6}\beta_0J^2v^{*2}\delta_t^3. \quad (40d)$$

The splitting errors for the two second-order Strang schemes are independent of the initial conditions.

3.2. Case of nucleation + condensation

Here, a case with constant nucleation rate and constant condensation growth in diameter is considered. A constant growth rate in diameter is not arbitrary at all. Eq. (7) gives the volume growth rate due to condensation as a piecewise function. The particle diameter growth rate function can be obtained easily through the sphere volume formula $v = \frac{\pi}{6}d_p^3$. The detailed

function can also be found in reference [31]. In fact, the particle diameter growth rate in the free molecular regime is independent of its size, which can be seen as a constant determined by environmental parameters (vapor pressure and temperature). In the following discussion, the particle diameter growth rate is assumed as G . In this subsection, the moments are defined with respect to the particle diameter d_p other than the volume $M_k = \int n(d_p) d_p^k dd_p$. No attempt is sought to distinguish the symbols for moments defined differently, with the hope to reduce new symbols and to bring no confusion.

The dynamic equations for moments due to pure nucleation (to generate particles with diameter d_p^* at rate J) can be easily obtained with the help of the Delta function (6), which are

$$\frac{dM_0^{(1)}}{dt} = J, \quad (41a)$$

$$\frac{dM_1^{(1)}}{dt} = Jd_p^*, \quad (41b)$$

$$\frac{dM_2^{(1)}}{dt} = Jd_p^{*2}, \quad (41c)$$

$$\frac{dM_3^{(1)}}{dt} = Jd_p^{*3}. \quad (41d)$$

The dynamic equations for moments due to condensation (at a constant diameter growth rate G) are

$$\frac{dM_0^{(2)}}{dt} = \int -\frac{\partial(nG)}{\partial d_p} dd_p = 0, \quad (42a)$$

$$\frac{dM_1^{(2)}}{dt} = \int -\frac{\partial(nG)}{\partial d_p} d_p dd_p = G \int n dd_p = GM_0^{(2)}, \quad (42b)$$

$$\frac{dM_2^{(2)}}{dt} = \int -\frac{\partial(nG)}{\partial d_p} d_p^2 dd_p = 2G \int n d_p dd_p = 2GM_1^{(2)}, \quad (42c)$$

$$\frac{dM_3^{(2)}}{dt} = \int -\frac{\partial(nG)}{\partial d_p} d_p^3 dd_p = 3G \int n d_p^2 dd_p = 3GM_2^{(2)}. \quad (42d)$$

Hence, the dynamic moments equations for nucleation and condensation are

$$\frac{dM_0}{dt} = J, \quad (43a)$$

$$\frac{dM_1}{dt} = Jd_p^* + GM_0, \quad (43b)$$

$$\frac{dM_2}{dt} = Jd_p^{*2} + 2GM_1, \quad (43c)$$

$$\frac{dM_3}{dt} = Jd_p^{*3} + 3GM_2. \quad (43d)$$

Assume the initial condition at $t = 0$ is $[M_0^*, M_1^*, M_2^*, M_3^*]$, then Eq. (43) has solution

$$M_0(t) = Jt + M_0^*, \quad (44a)$$

$$M_1(t) = \frac{1}{2}JGt^2 + (Jd_p^* + GM_0^*)t + M_1^*, \quad (44b)$$

$$M_2(t) = \frac{1}{3}JG^2t^3 + (Jd_p^* + GM_0^*)Gt^2 + (Jd_p^{*2} + 2GM_1^*)t + M_2^*, \quad (44c)$$

$$M_3(t) = \frac{1}{4}JG^3t^4 + (Jd_p^* + GM_0^*)G^2t^3 + \frac{3}{2}(Jd_p^{*2} + 2GM_1^*)Gt^2 + (Jd_p^{*3} + 3GM_2^*)t + M_3^*. \quad (44d)$$

On the other hand, the operator splitting can be used to solve Eqs. (41) and (42). Parallel to the analyses in Section 3.1, four splitting schemes are considered as follows.

(i) Lie scheme $I_{\text{nuc}} \rightarrow G_{\text{cond}}$ (Denoted as Lie1)

After solving Eqs. (41) and (42) sequentially, the final solution at $t = \delta_t$ is found as

$$M_0^{\text{OP}}(t) = Jt + M_0^*, \quad (45a)$$

$$M_1^{\text{OP}}(t) = JGt^2 + (Jd_p^* + GM_0^*)t + M_1^*, \quad (45b)$$

$$M_2^{\text{OP}}(t) = JG^2t^3 + (2Jd_p^* + GM_0^*)Gt^2 + (Jd_p^{*2} + 2GM_1^*)t + M_2^*, \quad (45c)$$

$$M_3^{\text{OP}}(t) = JG^3t^4 + (3Jd_p^* + GM_0^*)G^2t^3 + (3Jd_p^{*2} + 3GM_1^*)Gt^2 + (Jd_p^{*3} + 3GM_2^*)t + M_3^*. \quad (45d)$$

The splitting error is found by comparing the splitting solution (45) with the exact solution (44)

$$Er_{0(i)} = 0, \quad (46a)$$

$$Er_{1(i)} = \frac{1}{2}JG\delta_t^2, \quad (46b)$$

$$Er_{2(i)} = Jd_p^*G\delta_t^2 + \frac{2}{3}JG^2\delta_t^3, \quad (46c)$$

$$Er_{3(i)} = \frac{3}{2}Jd_p^{*2}G\delta_t^2 + 2JG^2d_p^*\delta_t^3 + \frac{3}{4}JG^3\delta_t^4. \quad (46d)$$

(ii) Lie scheme $G_{\text{cond}} \rightarrow I_{\text{nuc}}$ (Denoted as Lie2)

After solving Eqs. (42) and (41) sequentially, the final solution at $t = \delta_t$ is found as

$$M_0^{\text{OP}}(t) = Jt + M_0^*, \quad (47a)$$

$$M_1^{\text{OP}}(t) = (Jd_p^* + GM_0^*)t + M_1^*, \quad (47b)$$

$$M_2^{\text{OP}}(t) = G^2M_0^*t^2 + (Jd_p^{*2} + 2GM_1^*)t + M_2^*, \quad (47c)$$

$$M_3^{\text{OP}}(t) = G^3M_0^*t^3 + 3G^2M_1^*t^2 + (Jd_p^{*3} + 3GM_2^*)t + M_3^*. \quad (47d)$$

The splitting error is found by comparing the splitting solution (47) with the exact solution (44)

$$Er_{0(ii)} = 0, \quad (48a)$$

$$Er_{1(ii)} = -\frac{1}{2} JG\delta_t^2, \quad (48b)$$

$$Er_{2(ii)} = -Jd_p^* G\delta_t^2 - \frac{1}{3} JG^2\delta_t^3, \quad (48c)$$

$$Er_{3(ii)} = -\frac{3}{2} Jd_p^{*2} G\delta_t^2 - 2Jd_p^* G^2\delta_t^3 - \frac{1}{4} JG^3\delta_t^4. \quad (48d)$$

(iii) Strang scheme $\frac{1}{2}I_{\text{nuc}} \rightarrow G_{\text{cond}} \rightarrow \frac{1}{2}I_{\text{nuc}}$ (Denoted as Strang1)

After sequentially solving Eq. (41) with half time step $\delta_t/2$, then Eq. (42) with full time step, and then Eq. (41) with half step again, the final solution at $t = \delta_t$ is found as

$$M_0^{\text{OP}}(t) = Jt + M_0^*, \quad (49a)$$

$$M_1^{\text{OP}}(t) = \frac{1}{2} JGt^2 + (Jd_p^* + GM_0^*)t + M_1^*, \quad (49b)$$

$$M_2^{\text{OP}}(t) = \frac{1}{2} JG^2t^3 + (Jd_p^* + GM_0^*)Gt^2 + (Jd_p^{*2} + 2GM_1^*)t + M_2^*, \quad (49c)$$

$$M_3^{\text{OP}}(t) = \frac{1}{2} JG^3t^4 + \left(\frac{3}{2} Jd_p^* + GM_0^*\right)G^2t^3 + \left(\frac{3}{2} Jd_p^{*2} + 3GM_1^*\right)Gt^2 + (Jd_p^{*3} + 3GM_2^*)t + M_3^*. \quad (49d)$$

The splitting error is found by comparing the splitting solution (49) with the exact solution (44)

$$Er_{0(iii)} = 0, \quad (50a)$$

$$Er_{1(iii)} = 0, \quad (50b)$$

$$Er_{2(iii)} = \frac{1}{6} JG^2\delta_t^3, \quad (50c)$$

$$Er_{3(\text{iii})} = \frac{1}{2} J G^2 d_p^* \delta_t^3 + \frac{1}{4} J G^3 \delta_t^4. \quad (50d)$$

(iv) Strang scheme $\frac{1}{2}G_{\text{cond}} \rightarrow I_{\text{nuc}} \rightarrow \frac{1}{2}G_{\text{cond}}$ (Denoted as Strang2)

After sequentially solving Eq. (42) with half time step $\delta_t/2$, then Eq. (41) with full time step, and then Eq. (42) with half step again, the final solution at $t = \delta_t$ is found as

$$M_0^{\text{OP}}(t) = Jt + M_0^*, \quad (51a)$$

$$M_1^{\text{OP}}(t) = \frac{1}{2} J G t^2 + (J d_p^* + G M_0^*) t + M_1^*, \quad (51b)$$

$$M_2^{\text{OP}}(t) = \frac{1}{4} J G^2 t^3 + (J d_p^* + G M_0^*) G t^2 + (J d_p^{*2} + 2 G M_1^*) t + M_2^*, \quad (51c)$$

$$M_3^{\text{OP}}(t) = \frac{1}{8} J G^3 t^4 + \left(\frac{3}{4} J d_p^* + G M_0^*\right) G^2 t^3 + \left(\frac{3}{2} J d_p^{*2} + 3 G M_1^*\right) G t^2 + (J d_p^{*3} + 3 G M_2^*) t + M_3^*. \quad (51d)$$

The splitting error is found by comparing the splitting solution (51) with the exact solution (44)

$$Er_{0(\text{iv})} = 0, \quad (52a)$$

$$Er_{1(\text{iv})} = 0, \quad (52b)$$

$$Er_{2(\text{iv})} = -\frac{1}{12} J G^2 \delta_t^3, \quad (52c)$$

$$Er_{3(\text{iv})} = -\frac{1}{4} J G^2 d_p^* \delta_t^3 - \frac{1}{8} J G^3 \delta_t^4. \quad (52d)$$

To summarize, there is no splitting error for the number density (M_0), which is solely determined by nucleation. For M_1 to M_3 , all Lie schemes are of first order (globally), and the leading errors are of opposite sign between Lie1 and Lie2. For Strang schemes, the splitting error for M_1 is also zero, and the errors for M_2 and M_3 are of second order. In Lie1 and Strang1 scheme,

the splitting solutions overpredict the high order moments, since nucleation in the prestep produces more than real particles to grow in condensation. On the other hand, Lie2 and Strang2 underpredict the high order moments. Since the splitting errors are in closed form of the time step δ_t , it is possible to construct higher order splitting scheme with zero splitting error for all moments.

Furthermore, the leading order (with respect to δ_t) splitting errors for any two subsequent moments differ by a factor of d_p^* , which renders the relative errors (splitting errors divided by the corresponding moments values) comparable among various moments, since two subsequent moments also differ by a factor of the particles average diameter according to the moment definition. The splitting errors are independent of the initial conditions, i.e., M_0^* , M_1^* , M_2^* , and M_3^* , which seems strange at first glance. In fact, the splitting errors are anchored to the moment's evolution. Take the M_1 splitting error in the Lie1 scheme as example, the error contains the factor JG , while the leading order in the analytical solution for M_1 in Eq. (58) also contains exactly the same factor JG . Then only the time step δ_t matters when calculating the relative error. Generally speaking, the relative errors due to the operator splitting are small, almost independent of the nucleation rate and the condensational growth rate.

4. Numerical examples

4.1. Case of nucleation + coagulation

Corresponding to the error analyses in Section 3.1, the case of constant nucleation and constant coagulation are simulated with the OSMC. The coagulation kernel and nucleation rate are set as unit $\beta_0 = 1$, and $J = 1$. Initially particles are mono dispersed with unit volume size, and the number density is $M_0^* = 1$. Nucleated particles also have unit volume size.

Figure 2 shows the number density (M_0) simulated by the four schemes, compared against the analytical solution (19). All the simulations use fixed time step $\delta_t = 0.1$, with $N = 2000$ Monte Carlo particles. The OSMC results are averaged over dozens of independent simulations to eliminate the randomness. Lie1 scheme is found to systematically underpredict the number density, while Lie2 to overpredict. Strang1 gives result negligibly higher than the analytical solution. Strang2 slightly underpredicts the true solution. All these findings agree extremely well with the analyses in Section 3.1. It is also found that Strang1 shows better precision than Strang2, that is because nucleation rate is higher than the coagulation rate ($J = 1 > \frac{1}{2}\beta_0 M_0^* = \frac{1}{2}$), and the higher rate process nucleation is split in Strang1, as explained in Section 3.1 also.

Figure 3 compares M_2 among the four splitting schemes and the analytical solution (39). Lie1 is found to overestimate M_2 a little bit, while Lie2 to underestimate. The second-order Strang

schemes give result almost indistinguishable from the analytical solution. Although second-order Strang schemes give better results than the Lie schemes, the discrepancy is not so significant as that for M_0 in **Figure 2**.

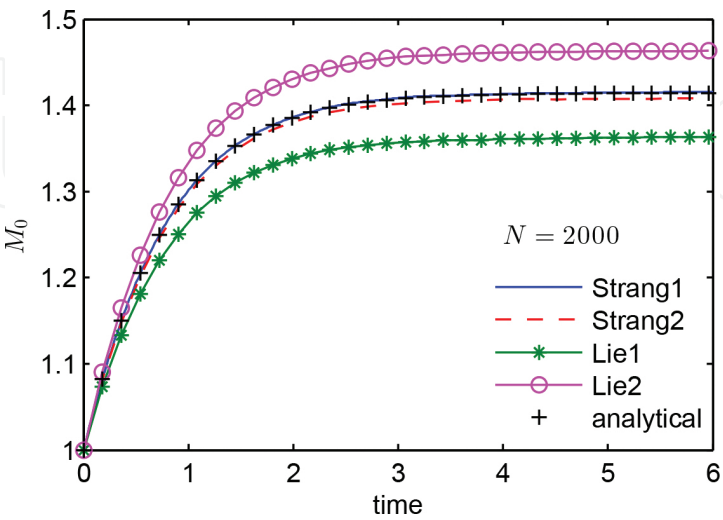


Figure 2. Evolution of total particle number density M_0 for constant kernel coagulation and constant rate nucleation ($\delta_t = 0.1$).

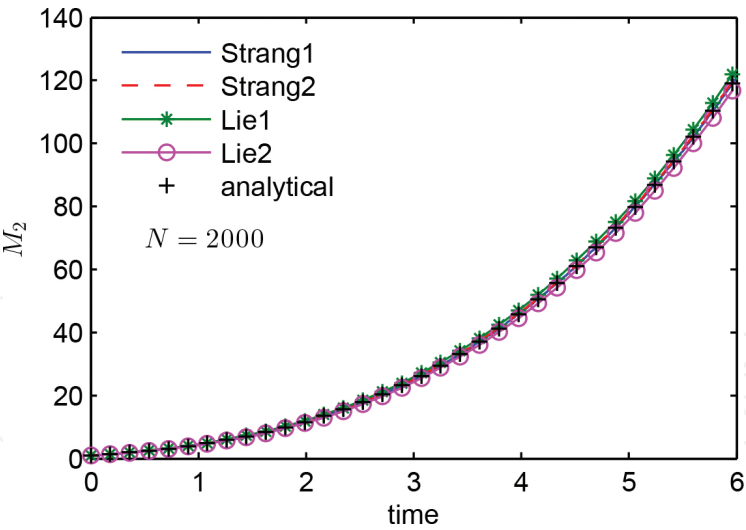


Figure 3. Evolution of moment M_2 in case of constant kernel coagulation and constant rate nucleation (moment is defined with respect to the particle volume).

Both the error analyses and the numerical examples above show that the second-order Strang schemes have much higher precision than the first-order Lie schemes. Within the OSMC, coagulation is the most computationally intensive part, and nucleation only makes a tiny fraction of the total time cost. The Strang1 scheme has much higher precision than the

Lie schemes, and only needs to integrate nucleation twice often as in the Lie schemes, where the computational overhead is negligible. Strang2 needs nearly double computational cost as Strang1, and Strang2 outperforms Strang1 in precision only if $\frac{1}{2}\beta_0 M_0^{*2} > J$. The best performance of Strang2 over Strang1 is to nearly double the precision when $\beta_0 M_0^{*2} \gg J$. Hence, Strang1 is the optimal splitting scheme taking both the numerical precision and cost into account.

Another general conclusion is that only M_0 needs to be concerned when considering the operator splitting errors. Operator splitting generally produces less errors for higher order moments. The M_2 splitting errors for the Strang schemes contain the factor of the critical volume of nucleated particles, which is generally very small. Taking a more practical numerical example of dibutyl phthalate (DBP) aerosol nucleation and coagulation in a turbulent mixing layer [26], the critical volume v^* is of order 10^{-27}m^3 , which renders very small splitting errors for high order moments. However, the splitting error for M_0 may be still significant, and needs proper splitting scheme combined with small enough time step to satisfy the given error criteria.

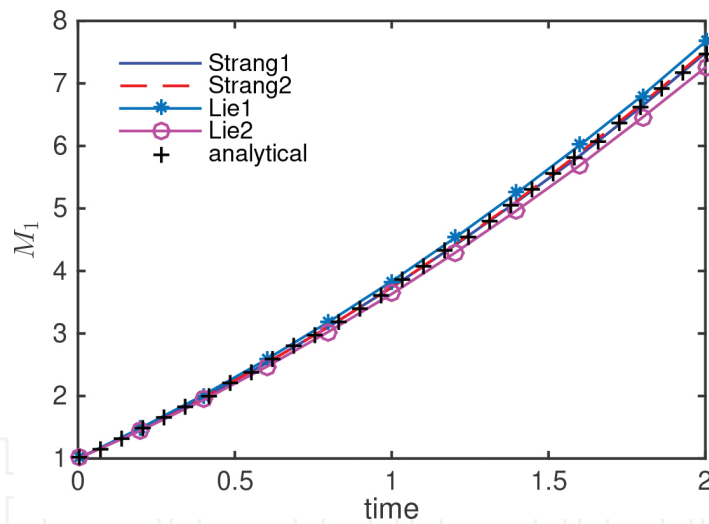


Figure 4. Evolution of moment M_1 in case of constant nucleation and constant condensational growth in diameter (moment is defined with respect to particle diameter).

4.2. Case of nucleation + condensation

Corresponding to the error analysis in Section 3.2, the case of constant nucleation and constant condensational growth in diameter is simulated with the OSMC. Both the nucleation rate and growth rate are set to unit $J = 1$ and $G = 1$. Initially particles are monodispersed with unit volume size, and the number density is $M_0^* = 1$. Nucleated particles also have unit volume size.

The time step is $\delta_t = 0.2$.

Figures 4 and 5 show the evolution of moments M_1 and M_3 , respectively, obtained by the four splitting schemes and the analytical solutions (44d). While M_3 directly represents the particle volume fraction (lacking a factor of $\pi/6$), M_1 do not have simple physical meaning, which may be used to define an average diameter through M_1/M_0 . For both M_1 and M_3 , the characteristics of splitting errors are quite similar. Lie1 is found to overpredict, while Lie2 to underpredict. The splitting errors for the two Strang schemes are very small, almost indistinguishable from the analytical solutions. According to the error analyses (49c) and (51c), the splitting errors for M_1 Strang schemes are zero. Close scrutiny shows in **Figure 4** that the numerical error is not actually zero, which actually arises from the ordinary differential equation (ODE) integration error, since a relatively large time step $\delta_t = 0.2$ is used. For a more appropriate smaller time step, both the ODE integration error and splitting error are much smaller. Comparing **Figures 4 and 5**, the relative errors are comparable for both M_1 and M_3 here. All the findings agree very well with the analyses in Section 3.2.

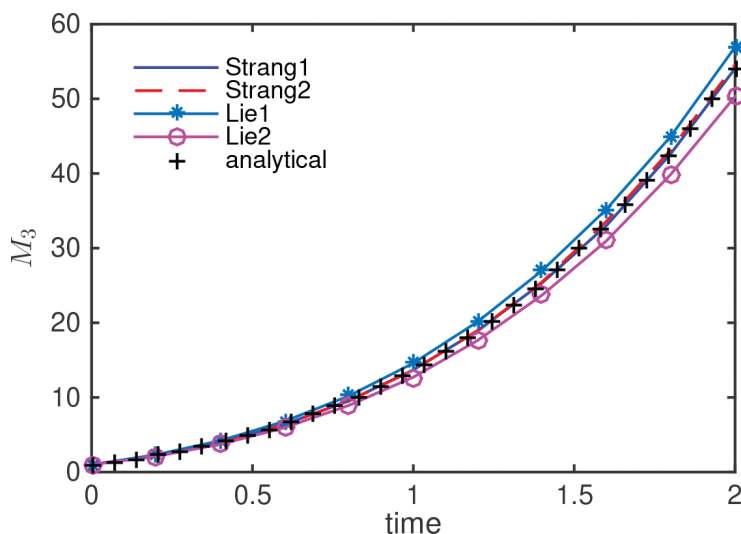


Figure 5. Evolution of moment M_3 (volume fraction missing a factor $\pi/6$) in case of constant nucleation and constant condensational growth in diameter (moment is defined with respect to particle diameter).

5. Conclusions

The operator splitting Monte Carlo (OSMC) method has been developed recently [26] for solving the population balance equation for aerosol dynamics. Within the OSMC, nucleation and surface growth are handled with deterministic means, while coagulation is simulated with a stochastic method (the Marcus-Lushnikov stochastic process). The deterministic and stochastic approaches in the algorithm are synthesized under the framework of operator splitting. The ultimate goal of the OSMC is to greatly improve the numerical efficiency and to

preserve the extraordinary flexibility and applicability of a stochastic method for aerosol dynamics simulation.

The OSMC has been validated through quite a few testing cases [26], and has also been used to simulate aerosol evolution under various conditions [32, 39]. However, the operator splitting errors in the OSMC have not been systematically investigated. Here, focused on two representative cases, i.e., constant nucleation and coagulation, and constant nucleation and condensation, the splitting errors for four splitting schemes (two Lie schemes and two Strang schemes) are analyzed rigorously, combined with concrete numerical examples.

The number density, one of the most important statistics for aerosol particles, is found to be underestimated for the Lie1 scheme ($I_{\text{nuc}} \rightarrow C_{\text{coag}}$), while overestimated for the Lie2 scheme ($C_{\text{coag}} \rightarrow I_{\text{nuc}}$). This is because coagulation rate would be overpredicted if nucleation is integrated before coagulation, hence the number density would be underpredicted as in Lie1, and vice versa. The second-order Strang schemes exhibit much better precision than the first-order Lie schemes. If coagulation rate is higher than the nucleation rate, Strang2 ($\frac{1}{2}C_{\text{coag}} \rightarrow I_{\text{nuc}} \rightarrow \frac{1}{2}C_{\text{coag}}$) has better precision than Strang1 ($\frac{1}{2}I_{\text{nuc}} \rightarrow C_{\text{coag}} \rightarrow \frac{1}{2}I_{\text{nuc}}$). Otherwise, Strang1 has better precision. However, for the OSMC, Strang1 is always the most preferable splitting scheme after taking the numerical cost and precision into account.

The operator splitting errors for the case of nucleation and condensation show that splitting errors for different moments (except for M_0) are comparable with each other. Splitting between nucleation and condensation has marginal impact on accurately predicting the moments. This is in great contrast to the prediction of M_0 , where splitting between nucleation and coagulation has significant impact.

To construct a good operator splitting scheme for the general case of nucleation, coagulation, and condensation the first priority is to consider the splitting error for M_0 . When a good scheme for splitting between nucleation and coagulation is constructed, splitting among other aerosol dynamic processes can be constructed at more freedom. The overall splitting errors are pretty much determined by the splitting error for M_0 .

The analyses not only provide sound theoretical bases for selecting the most efficient operator splitting scheme for the usage of the OSMC, but also shed lights on how to adopt operator splitting in other PBE solving methods, i.e., direct discretization, method of moments etc.

Acknowledgements

The financial support from the National Natural Science Foundation of China (No. 11402179 and 11572274) is greatly acknowledged. KZ would like to thank Dr. F. Bisetti at KAUST, Saudi Arabia, for his insightful discussion and valuable help. The financial support from the General Research Fund, Research Grants Council of the Hong Kong SAR, China (No.

PolyU 152663/16E) is also greatly appreciated for allowing the authors' further development and extension of this OSCM foundation method.

Author details

Kun Zhou^{1*} and Tat Leung Chan²

*Address all correspondence to: zhou.kun@wust.edu.cn

1 The State Key Laboratory of Refractories and Metallurgy, Wuhan University of Science and Technology, Wuhan, China

2 Department of Mechanical Engineering, The Hong Kong Polytechnic University, Kowloon, Hong Kong

References

- [1] R. L. Drake. A general mathematical survey of the coagulation equation. In G. M. Hidy and J. R. Brock, editors, *Topics in Current Aerosol Research*, Pergamon Press, New York, pp. 201–376, 1972.
- [2] D. Ramkrishna. *Population Balances: Theory and Applications to Particulate Systems in Engineering*. Academic Press, San Diego, 2000.
- [3] S. K. Friedlander. *Smoke, Dust, and Haze: Fundamentals of Aerosol Dynamics*. Oxford University Press, New York, second edition, 2000.
- [4] M. Smoluchowski. Drei Vorträge über Diffusion, Brownsche Bewegung und Koagulation von Kolloidteilchen. *Physikalische Zeitschrift*, 17:557–585, 1916.
- [5] D. J. Aldous. Deterministic and stochastic models for coalescence (aggregation and coagulation): a review of the mean-field theory for probabilists. *Bernoulli*, 5:3–48, 1999.
- [6] T. E. Ramabhadran, T. W. Peterson, and J. H. Seinfeld. Dynamics of aerosol coagulation and condensation. *AIChE Journal*, 220(5):840–851, 1976.
- [7] S. C. Davies, J. R. King, and J. A. D. Wattis. The Smoluchowski coagulation equations with continuous injection. *Journal of Physics A: Mathematical and Theoretical*, 32:7745–7763, 1999.
- [8] F. Gelbard, Y. Tambour, and J. H. Seinfeld. Sectional representations for simulating aerosol dynamics. *Journal of Colloid and Interface Science*, 760(2):541–556, 1980.
- [9] S. Rigopoulos and A. G. Jones. Finite-element scheme for solution of the dynamic population balance equation. *AIChE Journal*, 490(5):1127–1139, 2003.

- [10] A. H. Alexopoulos, A. I. Roussos, and C. Kiparissides. Part I: Dynamic evolution of the particle size distribution in particulate processes undergoing combined particle growth and aggregation. *Chemical Engineering Science*, 59(24):5751–5769, 2004.
- [11] S. Ganesan. An operator-splitting Galerkin/SUPG finite element method for population balance equations : stability and convergence. *ESAIM. Mathematical Modelling and Numerical Analysis*, 46(6):1447–1465, 2012.
- [12] J. Solsvik and H. A. Jakobsen. Evaluation of weighted residual methods for the solution of a population balance model describing bubbly flows: the least-squares, Galerkin, tau, and orthogonal collocation methods. *Industrial and Engineering Chemistry Research*, 52(45):15988–16013, 2013.
- [13] Y. P. Kim and J. H. Seinfeld. Simulation of multicomponent aerosol dynamics. *Journal of Colloid and Interface Science*, 149(2):425–449, 1991.
- [14] S. Kumar and D. Ramkrishna. On the solution of population balance equations by discretization-I. A fixed pivot technique. *Chemical Engineering Science*, 51(8):1311–1332, 1996.
- [15] S. Kumar and D. Ramkrishna. On the solution of population balance equations by discretization-II. A moving pivot technique. *Chemical Engineering Science*, 51(8):1333–1342, 1996.
- [16] J. Kumar, M. Peglow, G. Warnecke, and S. Heinrich. An efficient numerical technique for solving population balance equation involving aggregation, breakage, growth, and nucleation. *Powder Technology*, 182(1):81–104, 2008.
- [17] J. Kumar, M. Peglow, G. Warnecke, and S. Heinrich. The cell average technique for solving multi-dimensional aggregation population balance equations. *Computers and Chemical Engineering*, 32(8):1810–C1830, 2008.
- [18] S. E. Pratsinis. Simultaneous nucleation, condensation, and coagulation in aerosol reactors. *Journal of Colloid and Interface Science*, 124(2):416–427, 1988.
- [19] R. McGraw. Description of aerosol dynamics by the quadrature method of moments. *Aerosol Science and Technology*, 27(2):255–265, 1997.
- [20] D. L. Marchisio and R. O. Fox. Solution of population balance equations using the direct quadrature method of moments. *Journal of Aerosol Science*, 36(1):43–73, 2005.
- [21] M. Frenklach. Method of moments with interpolative closure. *Chemical Engineering Science*, 57(12):2229–2239, 2002.
- [22] M. Z. Yu, J. Z. Lin, and T. L. Chan. A new moment method for solving the coagulation equation for particles in Brownian motion. *Aerosol Science and Technology*, 42(9):705–713, 2008.
- [23] J. A. Shohat and J. D. Tamarkin. *The Problem of Moments*. American Mathematical Society, Providence, Rhode Island, revised edition, 1970.

- [24] Douglas L. Wright Jr. Numerical advection of moments of the particle size distribution in Eulerian models. *Journal of Aerosol Science*, 38:352–369, 2007.
- [25] D. T. Gillespie. An exact method for numerically simulating the stochastic coalescence process in a cloud. *Journal of the Atmospheric Sciences*, 32(10):1977–1989, 1975.
- [26] K. Zhou, Z. He, M. Xiao, and Z. Zhang. Parallel Monte Carlo simulation of aerosol dynamics. *Advances in Mechanical Engineering*, 2014: 435936, 2014.
- [27] K. Rajamani, W. T. Pate, and D. J. Kinneberg. Time-driven and event-driven Monte Carlo simulations of liquid-liquid dispersions: a comparison. *Industrial and Engineering Chemistry Fundamentals*, 25:746–752, 1986.
- [28] N. Ahmed, G. Matthies, and L. Tobiska. Finite element methods of an operator splitting applied to population balance equations. *Journal of Computational and Applied Mathematics*, 236(6):1604–1621, 2011.
- [29] N. Ahmed, G. Matthies, and L. Tobiska. Stabilized finite element discretization applied to an operator-splitting method of population balance equations. *Applied Numerical Mathematics*, 70(0):58–79, 2013.
- [30] F. Anker, S. Ganesan, V. John, and E. Schmeier. A comparative study of a direct discretization and an operator-splitting solver for population balance systems. *Computers and Chemical Engineering*, 75:95–104, 2015.
- [31] K. Zhou, A. Attili, A. Alshaarawi, and F. Bisetti. Simulation of aerosol nucleation and growth in a turbulent mixing layer. *Physics of Fluids*, 26:065106, 2014.
- [32] K. Zhou and Z. He. Monte Carlo simulation of aerosol evolution in a planar mixing layer. *International Journal of Numerical Methods for Heat & Fluid Flow*, 24(8):1769–1781, 2014.
- [33] R. I. McLachlan and G. R. W. Quispel. Splitting methods. *Acta Numerica*, 11:341–434, 2002.
- [34] V. I. Kalikmanov. *Nucleation Theory*. Springer, Dordrecht, The Netherlands, 2013.
- [35] R. Becker and W. Döring. Kinetische Behandlung der Keimbildung in übersättigten Dämpfen. *Annalen der Physik (Leipzig)*, 24(8):719–752, 1935.
- [36] B. D. Flury. Acceptance-reject ion sampling made easy. *SIAM Review*, 32(3):474–476, 1990.
- [37] G. S. Fishman. *Monte Carlo: Concepts, Algorithms, and Applications*. Springer-Verlag, New York, 1996.
- [38] H. Jia and K. Li. A third accurate operator splitting method. *Mathematical and Computer Modelling*, 53(1–2):387–396, 2011.
- [39] Z. He, K. Zhou, M. Xiao, and F. Wei. Simulation of soot size distribution in a counterflow flame. *International Journal of Chemical Reactor Engineering*, 12(2):1–7, 2014.