Cambridge Centre for Computational Chemical Engineering

University of Cambridge

Department of Chemical Engineering

Preprint ISSN 1473 – 4273

Direct Simulation and Mass Flow Stochastic Algorithms to Solve a Sintering-Coagulation Equation

Clive G. Wells and Markus Kraft [1]

submitted: 10th April 2003

[1] Department of Chemical Engineering
University of Cambridge
Pembroke Street
Cambridge CB2 3RA
UK
E-Mail: cgw11@cam.ac.uk, markus_kraft@cheng.cam.ac.uk

Preprint No. 10

c4e

Key words and phrases. Particle Size distribution, Monte Carlo Methods, Stochastic Processes, Sintering, Coagulation, Mass Flow.

Edited by
Cambridge Centre for Computational Chemical Engineering
Department of Chemical Engineering
University of Cambridge
Cambridge CB2 3RA
United Kingdom.

Fax: + 44 (0)1223 334796
E-Mail: c4e@cheng.cam.ac.uk
World Wide Web: http://www.cheng.cam.ac.uk/c4e/

Abstract

In this paper we develop an efficient stochastic method to solve the time evolution of a bivariate population balance equation which has been developed for modelling nano-particle dynamics. We have adapted the existing stochastic models used in the study of coagulation dynamics to solve a variant of the sintering-coagulation equation proposed by Xiong & Pratsinis. Hitherto stochastic models based on Markov jump processes have not taken into account the surface area evolution. We produce numerical results efficiently with the direct simulation and mass flow algorithms and study the convergence behaviour as the number of stochastic particles increases. We find a marked preference for using the mass flow algorithm to determine the higher order volume and area moments of the particle size distribution function. The computational efficiency of these algorithms is remarkable when compared to the sectional method that has been used previously to study this equation.

Contents

1 Introduction

Population balance models may be often applied to study the evolution of the macroscopic properties of a microscopic system. A particularly important application in the field of chemical engineering is that of nano-particle synthesis and evolution. Such nano-particles are produced in vast quantities by industry. Two of the most important industrial examples of nano-particle synthesis are the production of carbon blacks and powder ceramics. Both of these are produced in high temperature reactors. It is useful to develop modelling techniques that can be used to simulate the evolution of the macroscopic properties during the synthesis process. Particles of varying sizes and geometry are produced through the coagulation of smaller particles in the system. The simplest equation governing the evolution of the concentration of the particles in the system as a function of particle mass (or equivalently volumes) is the Smoluchowski equation [1]. This equation depends on the coagulation kernel, which describes the rate per unit volume at which any two particles of given masses coagulate. In some circumstances it is desirable to have additional information such as the evolution of the surface areas of the particles in the system. This bivariate evolution problem has recently received some attention. The Smoluchowski equation may be modified easily to incorporate the interaction due to surface areas by using a coagulation kernel that depends on the surface areas of the coagulating particles. Additionally surface areas evolve due to sintering, which for our purposes describes the tendency of a particle to change its morphology in such a way as to minimize its surface area. The development of models incorporating this surface area evolution include important contributions from Koch & Friedlander [2] and Xiong & Pratsinis [3].

Three main approaches have been developed in the study of the population balance equation in nano-particle synthesis and evolution problems. The first of these may be conveniently referred to as moment methods. Rather than solve for the Smoluchowski equation or its generalizations directly, one derives a set of ordinary differential equations relating various moments of the concentration. Frequently the moments of the population density are the physically interesting quantities, however when other information is required moment methods are usually inapplicable. Moment methods and their generalizations have been successfully applied to the bivariate evolution problem by Koch & Friedlander [2] who make particular reference to the application of their method to the synthesis of a silica (SiO_2) fume in a plug flow reactor. This method was developed by Kazakov & Frenklach [4] to study the formation and evolution of soot particles. In order to model both the free molecular regime and the continuum regime as the coagulation process evolved they took the coagulation moments to be the harmonic mean of the coagulation moments generated by using the kernels for both regimes. In the work of Kazakov & Frenklach

sintering was not considered, and the surface area not explicitly tracked, however the growth of fractal aggregates from spherical primary particles was modelled.

A bivariate method of moments has been implemented by Rosner & Pyykönnen [5] to model the synthesis of alumina (Al_2O_3) aggregates in a laminar flame reactor. In their work, Rosner and Pyykönnen solved for nine moments using a CFD generated temperature profile for the laminar flame and a coagulation kernel depending on an estimate of the diameter of the particles. Schwade & Roth [6] have used a moment method with a log normal *ansatz* for the particle size distribution to solve an inhomogeneous coagulation-sintering-diffusion-convection equation coupled to CFD code that solves the gas flow. This models the production of silicon particles from silane (SiH_4) in a wall heated aerosol reactor.

The second main approach to solving the population balance equation are referred to as sectional techniques. In this class of methods the configuration space of the particle volume (and surface area, if appropriate) is divided into a number of regions, within each region the concentration may be chosen to obey a simple rule, typically being constant or inversely proportional to the volume. One then computes the fluxes across neighbouring sections and solves a sparse matrix problem. The procedure should generate a solution to the Smoluchowski equation under successive refinement of the sections. Hounslow [7] developed this discretized population balance equation and used it to analyse the crystallization of nickel ammonium sulphate. The book by Ramkrishna [8] also sets out the method in some generality. Furthermore the sectional method has been used by Lindackers *et al.* [9] to model the formation and growth of silica particles.

The sectional method has been extended by Xiong & Pratsinis [3] to model the bivariate population balance equation tracking both volume and surface area of the aggregates. They used the sectional method to model the synthesis of titania (TiO_2) powder. In their subsequent paper Xiong *et al.* [10] applied their algorithm to the evolution of titania, silica and silica-doped titania powders. This model has been further improved in [11, 12, 13, 14] where the coagulation kernel has been modified to take account of the structure of the aggregates, the computational effort of the sectional method has been reduced and surface growth has been modelled. Despite the fact that this model offers more insight into the physics of particle formation and evolution it has not been widely used due to the immense numerical effort. Even recently [12], run times of up to 100 days on a PC for one simulation have been reported for a genuine bivariate sectional approach.

The third main class of methods to solve numerically the population balance equation are Monte Carlo methods. The first Monte Carlo approach is to model the microscopic physical processes involved that give rise to the population balance equation. This was the approach adopted by Gillespie [15]. The ideas behind this method are explained further by Ramkrishna [8]. Nano-particle synthesis and evo-

lution has been modelled using a bivariate Monte Carlo method for the continuum regime by Tandon & Rosner [16] and for the free molecular regime by Rosner & Yu [17]. Kruis *et al.* [18] compared several Monte Carlo methods and applied a new algorithm to the numerical investigation of the chemical reactions in coagulating droplets and the coating of particles with smaller particles. The main disadvantage of this first type of Monte Carlo simulation is the high computational effort necessary to reduce the statistic fluctuation in the computed solution, it is also rather difficult to justify convergence of the algorithm to the solution of the population balance equation purporting to describe the physical system.

The second type of Monte Carlo approach tackles the problem of solving the population balance equation from a mathematical angle rather than from a physical one. In this approach a stochastic process is generated in such a way that the sample paths form a solution to the population balance equation in the limit as a model parameter, known as the scaling parameter, increases without bound. Since a number of convergence proofs exist for these algorithms [19, 20, 21, 22] careful implementation of these algorithms will give numerical solutions with arbitrary accuracy. It is typical that convergence rates are extremely fast, and frequently of orders of magnitude faster that the sectional approach. At the forefront of the application of these results to chemical engineering are Eibeck & Wagner [22, 23, 24, 25], and Wagner [26] who have applied the methods to the study of coagulation and fragmentation of particulate systems. The methods of Eibeck & Wagner have been used by Goodson & Kraft [27] to develop a majorant kernel which they used to increase the efficiency of nano-particle coagulation simulations for a coagulation kernel involving an effective fractal dimension to account for the geometry of the aggregates in the ensemble. In Grosschmidt *et al.* [28] the authors used both a moment method and a stochastic model using the majorant kernel derived by Goodson & Kraft to model the formation of silica particles from a low pressure flame doped with a silane precursor. The same majorant kernel was used by Balthasar & Kraft [29] to model the formation of soot particles from a realistic premixed $C_2H_2/O_2/Ar$ laminar flame model. They compared the stochastic results with some results obtained from the moment method.

The paper is organized as follows. In Sect. 2.1 we present a modification of the sintering-coagulation equation proposed by Xiong & Pratsinis [3]. After replacing the sintering term by a finite difference we establish the weak form of the modified equation. In Sect. 2.2 we form the stochastic method that solves this sintering-coagulation equation. We assume convergence of the stochastic algorithm, and believe that this should be straightforward to prove, though do not present a proof in this paper. We make use of a majorant kernel in Sect. 2.3 in order to dramatically improve the efficiency of the algorithm. This algorithm, which we refer to as the direct simulation algorithm (DSA) is written out in algorithmic terms in Sect. 2.4. We

make some comments on the efficient selection of particles according to a probability distribution, and on other techniques that are useful when implementing the algorithm. In Sect. 3 we write the sintering-coagulation equation in terms of the mass density rather than the number density. This leads to a bivariate formulation of the mass flow algorithm (MFA) originally introduced by Eibeck & Wagner [24]. Sect. 4 is dedicated to studying the numerical results obtained from the implementations of both the direct simulation and mass flow algorithms. The detailed comparison of the methods is presented in Sect. 4.1. We find rapid convergence of both algorithms, and present a set of graphs and tables showing the error and computational times when one varies the number of stochastic particles in the simulations. In Sect. 4.2 we study the dependence of our results on the area step size parameter that was introduced when the area derivative in the original sintering-coagulation equation was replaced by a finite difference. Finally we draw our conclusions in Sect. 5.

2 Modelling

2.1 The Weak Form of the Sintering-Coagulation Equation

A decade ago Xiong & Pratsinis [3] proposed a modification of the Smoluchowski equation that also modelled the sintering phenomenon. We refer to such extensions as sintering-coagulation equations, they determine the temporal evolution of the particle size distribution function $n_t(\mathbf{r}, v, a)$, i.e., the number density of particles in space, volume and surface area. In this paper we will assume that the particle distribution is spatially homogeneous and therefore there is no dependence on the position vector $\mathbf{r}$. Under these conditions a corrected version of the equation proposed by Xiong & Pratsinis is given by

$$\begin{aligned}\frac{\partial n_t(v,a)}{\partial t} &= \frac{\partial}{\partial a}\left(\frac{1}{t_0}\left[a - a_0\left(\frac{v}{v_0}\right)^{\frac{2}{3}}\right] n_t(v,a)\right) \\ &+ \frac{1}{2}\int_{v_0}^{v}\int_{a_0\left(\frac{v'}{v_0}\right)^{\frac{2}{3}}}^{\frac{a_0 v'}{v_0}} \mathbb{1}_{H_{\frac{v'}{v_0},\frac{v-v'}{v_0}}}\left(\frac{a}{a_0}\right)\beta_{v,v'}(a', a-a')n_t(v',a')n_t(v-v',a-a')\,\mathrm{d}a'\mathrm{d}v' \\ &- n_t(v,a)\int_{v_0}^{\infty}\int_{a_0\left(\frac{v'}{v_0}\right)^{\frac{2}{3}}}^{\frac{a_0 v'}{v_0}} \beta_{v,v'}(a,a')n_t(v',a')\,\mathrm{d}a'\mathrm{d}v' \qquad (1)\end{aligned}$$

where we have set $H_{c,d} = [c^{\frac{2}{3}} + d^{\frac{2}{3}}, \infty)$. The characteristic sintering time is denoted by t_0, which we take to be constant, however it is easy to extend our results to the case when t_0 is a function of volume and surface area; in particular if we take $t_0 \propto (v/a)^4$ then we would recover a corrected model of Xiong & Pratsinis. In contrast to

Koch & Friedlander [2] and inconsistent with other equations in their paper, Xiong & Pratsinis [3] took the characteristic sintering time outside of the surface area derivative. In doing so one finds that this does not reproduce the Smoluchowski equation properly when integrated over the surface area. The quantities a_0, v_0 and m_0 in Eq. (1) are the surface area, volume and mass of the primary particles, i.e., the smallest particles in the system at the starting time $t = 0$. The coagulation kernel $\beta_{v,v'}(a, a')$ determines the rate per unit volume at which particles coagulate and is given by

$$\beta_{v,v'}(a, a') = \left(\frac{kTa_0^2}{2\pi m_0}\right)^{\frac{1}{2}} K(x, x') \tag{2}$$

where we have defined

$$K(x, x') = \left(\frac{1}{\nu} + \frac{1}{\nu'}\right)^{\frac{1}{2}} \left[(s(\nu)\sigma)^{\frac{1}{2}} + (s(\nu')\sigma')^{\frac{1}{2}}\right]^2 \tag{3}$$

with $x = (\nu, \sigma) = (v/v_0, a/a_0)$ and $s(\nu)$ is the surface area accessibility function given by

$$s(\nu) = \lambda_1 \nu^{\alpha-1} + \lambda_2; \qquad \lambda_1 = 2^{1-\alpha}(D_S - 2); \qquad \lambda_2 = 3 - D_S. \tag{4}$$

We have written $D_S \in [2, 3]$ for the surface fractal dimension and $\alpha \in [0, 1]$ for the surface area scaling factor. The temperature is denoted by T and k is Boltzmann's constant.

Throughout this paper we will impose the condition $n_t(v, a) = 0$ for all pairs (v, a) such that $a < a_0(v/v_0)^{\frac{2}{3}}$ or $a > a_0 v/v_0$. Physically this implements the constraint that the surface area of each particle lies between that of a perfect sphere and an uncoalesced aggregate of primary particles. In particular if $a_0(v'/v_0)^{\frac{2}{3}} \leqslant a'$ and $a_0((v - v')/v_0)^{\frac{2}{3}} \leqslant a - a'$ then $a_0[(v'/v_0)^{\frac{2}{3}} + ((v - v')/v_0)^{\frac{2}{3}}] \leqslant a$ and it follows that with our condition on $n_t(v, a)$ the indicator function in Eq. (1) performs no useful purpose.

We will implement stochastic algorithms that make use of Markov jump processes. One approach to implementation is to replace the derivative with respect to surface area in Eq. (1) by a finite difference. Let us write the differential-integral equation in terms of the dimensionless quantities x and $\tau = t/t_0$. The population density function is also expressed in terms of the dimensionless quantity $N_\tau(x)$ where

$$N_\tau(x) = t_0 a_0^2 v_0 \left(\frac{kT}{2\pi m_0}\right)^{\frac{1}{2}} n_t(v, a) \tag{5}$$

and a finite difference step size parameter Λ is introduced which determines the discretization of the surface area variable σ. The equation for which we will construct

a stochastic model is given by

$$\begin{aligned}\frac{\partial N_\tau(x)}{\partial \tau} &= \rho_\Lambda(x+\Lambda)N_\tau(x+\Lambda) - \rho_\Lambda(x)N_\tau(x) \\ &\quad + \frac{1}{2}\int_1^\nu \int_{\nu'^{\frac{2}{3}}}^{\nu'} K(x-x',x')N_\tau(x')N_\tau(x-x')\,\mathrm{d}^2x' \\ &\quad - \int_1^\infty \int_{\nu'^{\frac{2}{3}}}^{\nu'} K(x,x')N_\tau(x)N_\tau(x')\,\mathrm{d}^2x'; \qquad (6)\end{aligned}$$

$$\rho_\Lambda(x) = \begin{cases}\Lambda^{-1}\left(\sigma - \nu^{\frac{2}{3}}\right) & \text{if } \nu^{\frac{2}{3}} + \Lambda \leqslant \sigma \leqslant \nu, \\ 0 & \text{otherwise.}\end{cases} \qquad (7)$$

As a small abuse of notion we have defined expressions such as $x+\Lambda$ to be equal to $(\nu, \sigma+\Lambda)$. It will be convenient to define the region in configuration space given by

$$\mathcal{R} = \{(p,q) \in \mathbb{R}^2 \mid 1 \leqslant p^{\frac{2}{3}} \leqslant q \leqslant p\}. \qquad (8)$$

It may be observed that $\operatorname{supp}\rho_\Lambda \subseteq \mathcal{R} + \Lambda$ and $\operatorname{supp} N_\tau \subseteq \mathcal{R}$, since as previously remarked, each particle in the system has a surface area that lies between $a_0\nu^{\frac{2}{3}}$ and $a_0\nu$ where $v_0\nu$ is the particle's volume.

We construct next a weak form of this equation. Let $\varphi : \mathcal{R} \to \mathbb{R}$ be a bounded continuous function on $\mathcal{R}$. Integrate the product of Eq. (6) and $\varphi(x)$ over $\mathcal{R}$ to obtain

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}\tau}\int_{\mathcal{R}} \varphi(x)N_\tau(x)\,\mathrm{d}^2x &= \int_{\mathcal{R}} \left\{\rho_\Lambda(x+\Lambda)N_\tau(x+\Lambda) - \rho_\Lambda(x)N_\tau(x)\right\}\varphi(x)\,\mathrm{d}^2x \\ &\quad + \frac{1}{2}\int_{\mathcal{R}} \mathrm{d}^2x \int_1^\nu \int_{\nu'^{\frac{2}{3}}}^{\nu'} \varphi(x)K(x-x',x')N_\tau(x')N_\tau(x-x')\,\mathrm{d}^2x' \\ &\quad - \int_{\mathcal{R}}\int_{\mathcal{R}} \varphi(x)K(x,x')N_\tau(x)N_\tau(x')\,\mathrm{d}^2x\,\mathrm{d}^2x'. \qquad (9)\end{aligned}$$

At this point it is useful to write $N_\tau(x) = \mathbb{1}_{\mathcal{R}}(x)N_\tau(x)$, the second term on the RHS may be rewritten as

$$\frac{1}{2}\int_{\mathbb{R}^2}\int_{\mathbb{R}^2} \psi(x,x')\mathbb{1}_{\mathcal{R}}(x)\mathbb{1}_{\mathcal{R}}(x')\mathbb{1}_{\mathcal{R}}(x-x')\,\mathrm{d}^2x\,\mathrm{d}^2x' \qquad (10)$$

where

$$\psi(x,x') = K(x-x',x')N_\tau(x')N_\tau(x-x'). \qquad (11)$$

Expression (10) can be rewritten using $x \mapsto x+x'$ as

$$\frac{1}{2}\int_{\mathbb{R}^2}\int_{\mathbb{R}^2} \psi(x+x',x')\mathbb{1}_{\mathcal{R}}(x+x')\mathbb{1}_{\mathcal{R}}(x')\mathbb{1}_{\mathcal{R}}(x)\,\mathrm{d}^2x\,\mathrm{d}^2x' = \frac{1}{2}\int_{\mathcal{R}}\int_{\mathcal{R}} \psi(x+x',x')\,\mathrm{d}^2x\,\mathrm{d}^2x'. \qquad (12)$$

It follows that the weak form of Eq. (6) can be rewritten as

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}\tau}\int_{\mathcal{R}}\varphi(x)N_\tau(x)\,\mathrm{d}^2x &= \int_{\mathcal{R}+\Lambda}\rho_\Lambda(x)N_\tau(x)\varphi(x-\Lambda)\,\mathrm{d}^2x-\int_{\mathcal{R}}\rho_\Lambda(x)N_\tau(x)\varphi(x)\,\mathrm{d}^2x\\ &+\frac{1}{2}\int_{\mathcal{R}}\int_{\mathcal{R}}\left\{\varphi(x+x')-\varphi(x)-\varphi(x')\right\}K(x,x')N_\tau(x)N_\tau(x')\,\mathrm{d}^2x\,\mathrm{d}^2x'. \qquad (13)\end{aligned}$$

Notice that

$$\begin{aligned}\int_{\mathcal{R}+\Lambda}\rho_\Lambda(x)N_\tau(x)\varphi(x-\Lambda)\,\mathrm{d}^2x &= \int_{\mathcal{R}+\Lambda}\rho_\Lambda(x)N_\tau(x)\mathbb{1}_{\mathcal{R}}(x)\varphi(x-\Lambda)\,\mathrm{d}^2x \qquad (14)\\ &= \int_{\mathcal{R}}\mathbb{1}_{\mathcal{R}+\Lambda}(x)\rho_\Lambda(x)N_\tau(x)\varphi(x-\Lambda)\,\mathrm{d}^2x \qquad (15)\\ &= \int_{\mathcal{R}}\rho_\Lambda(x)N_\tau(x)\varphi(x-\Lambda)\,\mathrm{d}^2x \qquad (16)\end{aligned}$$

since $\operatorname{supp}\rho_\Lambda\subseteq\mathcal{R}+\Lambda$.

Take $\mathcal{M}_b(\mathcal{R})$ to be the non-negative Borel measures on $\mathcal{R}$. We will call $P\in[0,\infty)\times\mathcal{M}_b(\mathcal{R})$ a measure-valued solution to the sintering-coagulation equation with initial distribution P_0 if $P(0,\cdot)=P_0$ and

$$\begin{aligned}\int_{\mathcal{R}}\varphi(x)P(\tau,\mathrm{d}x) &= \int_{\mathcal{R}}\varphi(x)P(0,\mathrm{d}x)+\int_0^\tau\mathrm{d}s\int_{\mathcal{R}}\left\{\varphi(x-\Lambda)-\varphi(x)\right\}\rho_\Lambda(x)P(s,\mathrm{d}x)\\ &+\frac{1}{2}\int_0^\tau\mathrm{d}s\int_{\mathcal{R}}\int_{\mathcal{R}}\left\{\varphi(x+x')-\varphi(x)-\varphi(x')\right\}K(x,x')P(s,\mathrm{d}x)P(s,\mathrm{d}x') \qquad (17)\end{aligned}$$

holds for all bounded continuous functions φ defined on $\mathcal{R}$ and for all $\tau>0$.

2.2 The Stochastic Coalescent

In this section we propose a sequence of measure-valued random variables that converge in distribution to a solution of Eq. (17). Let $V>0$ be a (large) fixed real number, define

$$\mathbb{S}_V^N=\left\{p\in\mathcal{M}_b(\mathcal{R})\;\middle|\;p=\frac{1}{N}\sum_{i=1}^n\delta_{x_i}^2,\ \int_{\mathcal{R}}\nu\,p(\mathrm{d}x)\leqslant V\right\} \qquad (18)$$

for each value of the scaling parameter N. In this formula $\delta_{x_i}^2$ is the Dirac point measure defined by $\delta_{x_i}^2(\Gamma)=\mathbb{1}_\Gamma(x_i)$. For any sequence of $U_\tau^N\in\mathbb{S}_V^N$ define the measure $\mu_\tau^N\in\mathcal{M}_b(\mathcal{R}\times\mathcal{R})$ by

$$\mu_\tau^N(\Gamma_1\times\Gamma_2)=U_\tau^N(\Gamma_1)U_\tau^N(\Gamma_2)-\frac{1}{N}U_\tau^N(\Gamma_1\cap\Gamma_2). \qquad (19)$$

Therefore $N^2\mu_\tau^N(\Gamma_1,\Gamma_2)$ is the number of distinct pairs of unordered indices i,j with $(x_i,x_j)\in\Gamma_1\times\Gamma_2$. With this interpretation it is clear that μ_τ^N is a non-negative Borel measure.

We will produce a sequence, the stochastic coalescent of measure-valued random variables U_τ^N in $\mathbb{S}_V^N$ which has the property that the following is a martingale

$$\begin{aligned} M_\varphi^N(\tau) &= \int_{\mathcal{R}}\varphi(x)U_\tau^N(\mathrm{d}x)-\int_{\mathcal{R}}\varphi(x)U_0^N(\mathrm{d}x) \\ &\quad -\int_0^\tau \mathrm{d}s\int_{\mathcal{R}}\left\{\varphi(x-\Lambda)-\varphi(x)\right\}\rho_\Lambda(x)U_s^N(\mathrm{d}x) \\ &\quad -\frac{1}{2}\int_0^\tau \mathrm{d}s\int_{\mathcal{R}}\int_{\mathcal{R}}\left\{\varphi(x+x')-\varphi(x)-\varphi(x')\right\}K(x,x')\mu_s^N(\mathrm{d}x,\mathrm{d}x') \end{aligned} \tag{20}$$

with zero as an accumulation point (where the notion of convergence is of convergence in distribution).

It should be noted that

$$\begin{aligned} &\left|\frac{1}{2}\int_{\mathcal{R}}\int_{\mathcal{R}}\left\{\varphi(x+x')-\varphi(x)-\varphi(x')\right\}K(x,x')U_s^N(\mathrm{d}x)U_s^N(\mathrm{d}x')\right. \\ &\qquad\left.-\frac{1}{2}\int_{\mathcal{R}}\int_{\mathcal{R}}\left\{\varphi(x+x')-\varphi(x)-\varphi(x')\right\}K(x,x')\mu_s^N(\mathrm{d}x,\mathrm{d}x')\right| \\ =\;& \frac{1}{2N}\left|\int_{\mathcal{R}}\{\varphi(2x)-2\varphi(x)\}K(x,x)U_s^N(\mathrm{d}x)\right| \\ \leqslant\;& \frac{3\|\varphi\|_\infty}{2N}\int_{\mathcal{R}}K(x,x)U_s^N(\mathrm{d}x) \qquad (21)\\ \leqslant\;& \frac{3\times 2^{\frac{7}{2}}\|\varphi\|_\infty}{N}\int_{\mathcal{R}}\nu^{\frac{1}{2}}U_s^N(\mathrm{d}x) \qquad (22)\\ \leqslant\;& \frac{3\times 2^{\frac{7}{2}}\|\varphi\|_\infty V}{N}\to 0 \qquad (23) \end{aligned}$$

as $N\to\infty$. We have used $K(x,x)\leqslant 2^{\frac{9}{2}}\nu^{\frac{1}{2}}$ for $x\in\mathcal{R}$. Accordingly we assume that if $U_\tau^{N_k}\Rightarrow P(\tau,\cdot)$ (convergence in distribution) and $M_\varphi^{N_k}(\tau)\Rightarrow 0$ as $k\to\infty$ then P is almost surely a solution to the sintering-coagulation equation Eq. (17). The purpose of this paper is not to present an appropriate convergence proof, however Norris [19, 20] and Eibeck & Wagner [22, 25] provide some important results in this direction that we believe should generalize easily to cover the situation we are studying.

Define the jump operators for any measure $p\in\mathcal{M}_b(\mathcal{R})$ by

$$J_c(p,x,x') = p+\frac{1}{N}\left(\delta_{x+x'}^2-\delta_x^2-\delta_{x'}^2\right); \tag{24}$$

$$J_s(p,x) = p+\frac{1}{N}\left(\delta_{x-\Lambda}^2-\delta_x^2\right). \tag{25}$$

For each $p \in \mathcal{M}_b(\mathcal{R})$ we will use the notation that $\Phi(p) = \int_{\mathcal{R}} \varphi(x) p(\mathrm{d}x)$. Consequently the generator associated with the martingale expression Eq. (20) is found to be

$$\begin{aligned}\mathcal{G}^N(\Phi)(U_s^N) &= N \int_{\mathcal{R}} \rho_\Lambda(x)[\Phi(J_s(U_s^N, x)) - \Phi(U_s^N)] U_s^N(\mathrm{d}x) \\ &\quad + \frac{N}{2} \int_{\mathcal{R}\times\mathcal{R}} [\Phi(J_c(U_s^N, x, x')) - \Phi(U_s^N)] K(x, x') \mu_s^N(\mathrm{d}x, \mathrm{d}x'). \end{aligned} \tag{26}$$

Next we introduce a majorant kernel $\widehat{K}(x, x')$, this is a symmetric positive definite upper bound on K, i.e., $K(x, x') \leqslant \widehat{K}(x, x')$ for all $x, x' \in \mathcal{R}$. Although Eq. (26) has no dependence on the majorant kernel it is useful to write it in the form

$$\begin{aligned}\mathcal{G}^N(\Phi)(U_s^N) &= N \int_{\mathcal{R}} \rho_\Lambda(x)[\Phi(J_s(U_s^N, x)) - \Phi(U_s^N)] U_s^N(\mathrm{d}x) \\ &\quad + \frac{N}{2} \int_{\mathcal{R}\times\mathcal{R}} \left\{ \frac{K(x, x')}{\widehat{K}(x, x')} [\Phi(J_c(U_s^N, x, x')) - \Phi(U_s^N)] \right. \\ &\quad \left. + \left(1 - \frac{K(x, x')}{\widehat{K}(x, x')}\right) [\Phi(U_s^N) - \Phi(U_s^N)] \right\} \widehat{K}(x, x') \mu_s^N(\mathrm{d}x, \mathrm{d}x'). \end{aligned} \tag{27}$$

It follows that $M_\varphi^N(\tau)$ is a martingale in τ for all bounded continuous functions $\varphi : \mathcal{R} \to \mathbb{R}$ if the time evolution of U_τ^N obeys the differential relationship

$$\mathbb{E}\left(U_{\tau+\mathrm{d}\tau}^N \mid U_\tau^N\right) - U_\tau^N = \mathbb{E}\left((\mathcal{P}^N(U_\tau^N) - U_\tau^N)\varrho(U_\tau^N) \mid U_\tau^N\right) \mathrm{d}\tau \tag{28}$$

for $\mathrm{d}\tau \geqslant 0$ and where the transition map $\mathcal{P}^N$ is defined by

$$\Phi((\mathcal{P}^N(U_\tau^N) - U_\tau^N)\varrho(U_\tau^N)) = \mathcal{G}^N(\Phi) U_\tau^N. \tag{29}$$

Accordingly we take

$$\varrho(U_s^N) = N \int_{\mathcal{R}} \rho_\Lambda(x) U_s^N(\mathrm{d}x) + \frac{N}{2} \int_{\mathcal{R}\times\mathcal{R}} \widehat{K}(x, x') \mu_s^N(\mathrm{d}x, \mathrm{d}x'); \tag{30}$$

$$\begin{aligned}\mathcal{P}^N(U_s^N) &= \frac{N}{\varrho(U_s^N)} \int_{\mathcal{R}} \rho_\Lambda(x) J_s(U_s^N, x) U_s^N(\mathrm{d}x) \\ &+ \frac{N}{2\varrho(U_s^N)} \int_{\mathcal{R}\times\mathcal{R}} \left\{ \frac{K(x, x')}{\widehat{K}(x, x')} J_c(U_s^N, x, x') + \left(1 - \frac{K(x, x')}{\widehat{K}(x, x')}\right) U_s^N \right\} \widehat{K}(x, x') \mu_s^N(\mathrm{d}x, \mathrm{d}x'). \end{aligned} \tag{31}$$

In our analysis so far we have assumed only that $U_s^N \in \mathcal{M}_b(\mathcal{R})$. If furthermore we

assume that $U_s^N \in \mathbb{S}_V^N$ then we find the explicit representations:

$$\varrho(U_s^N) = \sum_{i=1}^n \rho_\Lambda(x_i) + \frac{1}{2N} \sum_{\substack{i,j=1 \\ i\neq j}}^n \widehat{K}(x_i, x_j); \tag{32}$$

$$\begin{aligned} \mathcal{P}^N(U_s^N) &= \frac{1}{\varrho(U_s^N)} \sum_{i=1}^n \rho_\Lambda(x_i) J_s(U_s^N, x_i) \\ &+ \frac{1}{2N\varrho(U_s^N)} \sum_{\substack{i,j=1 \\ i\neq j}}^n \left\{ \frac{K(x_i,x_j)}{\widehat{K}(x_i,x_j)} J_c(U_s^N, x_i, x_j) + \left(1 - \frac{K(x_i,x_j)}{\widehat{K}(x_i,x_j)}\right) U_s^N \right\} \widehat{K}(x_i, x_j). \end{aligned} \tag{33}$$

Notice that in general $\mathcal{P}^N$ does not map $\mathbb{S}_V^N$ to $\mathbb{S}_V^N$. In order to generate sample paths in $\mathbb{S}_V^N$ we make use of the fact that U_s^N is a random variable. This allows us to keep U_s^N within $\mathbb{S}_V^N$. Since $\mathcal{P}^N$ is inside a conditional expectation in Eq. (28), we may achieve this by making a suitable probabilistic interpretation of the transition map $\mathcal{P}^N$.

The sample paths are generated by the following rule: given a state U_τ^N, generate an exponentially distributed waiting time $\widehat{\tau}$ with parameter $\varrho(U_\tau^N)$, then chose $U_{\tau+\widehat{\tau}}^N$ such that $\mathbb{E}(U_{\tau+\widehat{\tau}}^N \mid U_\tau^N) = \mathbb{E}(\mathcal{P}^N U_\tau^N \mid U_\tau^N)$. For all $s \in [0, \widehat{\tau})$ we set $U_s^N = U_\tau^N$. To see that this generates appropriate sample paths notice that the probability that the waiting time $\widehat{\tau}$ is within $\mathrm{d}\tau$ of the current time is $\mathbb{P}(0 < \widehat{\tau} < \mathrm{d}\tau) = \varrho(U_\tau^N)\,\mathrm{d}\tau$, therefore

$$\mathbb{E}\left(U_{\tau+\mathrm{d}\tau}^N \mid U_\tau^N\right) = \mathbb{E}\left(\mathcal{P}^N(U_\tau^N) \mid U_\tau^N\right) \varrho(U_\tau^N)\,\mathrm{d}\tau + U_\tau^N(1 - \varrho(U_\tau^N)\,\mathrm{d}\tau) \tag{34}$$

which may be rearranged to give Eq. (28).

2.3 Choice of Majorant Kernel

In this section we establish a simple upper bound on $K(x, x')$, this upper bound is the majorant kernel $\widehat{K}(x, x')$. We will need the inequalities $(a+b)^{\frac{1}{2}} \leqslant a^{\frac{1}{2}} + b^{\frac{1}{2}}$ which is verified by squaring both sides, and $(a^{\frac{1}{2}} + b^{\frac{1}{2}})^2 \leqslant (a^{\frac{1}{2}} + b^{\frac{1}{2}})^2 + (a^{\frac{1}{2}} - b^{\frac{1}{2}})^2 = 2(a+b)$. Accordingly

$$K(x,x') = \left(\frac{1}{\nu} + \frac{1}{\nu'}\right)^{\frac{1}{2}} \left[\left(\lambda_1 \sigma \nu^{\alpha-1} + \lambda_2 \sigma\right)^{\frac{1}{2}} + \left(\lambda_1 \sigma \nu'^{\alpha-1} + \lambda_2 \sigma'\right)^{\frac{1}{2}}\right]^2 \tag{35}$$

$$\leqslant 2\left(\frac{1}{\nu^{\frac{1}{2}}} + \frac{1}{\nu'^{\frac{1}{2}}}\right)\left(\lambda_1 \sigma \nu^{\alpha-1} + \lambda_2 \sigma + \lambda_1 \sigma' \nu'^{\alpha-1} + \lambda_2 \sigma'\right). \tag{36}$$

Take the RHS of Ineq. (36) to be the definition of $\widehat{K}(x,x')$. Let us define

$$\theta(x) = \lambda_1\sigma\nu^{\alpha-3/2} + \lambda_2\sigma\nu^{-\frac{1}{2}} \qquad \text{and} \qquad \psi(x) = \lambda_1\sigma\nu^{\alpha-1} + \lambda_2\sigma. \tag{37}$$

It is simple to express the majorant kernel in terms of these quantities, thus:

$$\widehat{K}(x,x') = 2\left(\theta(x) + \psi(x)\nu'^{-\frac{1}{2}} + \psi(x')\nu^{-\frac{1}{2}} + \theta(x')\right). \tag{38}$$

Set

$$\varrho_c = \frac{1}{2N}\sum_{\substack{i,j=1\\ i\neq j}}^{n}\widehat{K}(x_i,x_j) = \frac{2(n-2)}{N}\sum_{i=1}^{n}\theta(x_i) + \frac{2}{N}\sum_{j=1}^{n}\nu_j^{-\frac{1}{2}}\sum_{i=1}^{n}\psi(x_i). \tag{39}$$

One of the computational advantages of using a majorant kernel is it takes $O(1)$ operations to update ϱ_c if a new particle is added or an existing particle removed from the system. The corresponding computation using the coagulation kernel $K(x,x')$ instead of its majorant takes $O(N)$ operations. Define the joint probability densities for $i \neq j$ by

$$p_{ij}^{(1)} = \frac{\theta(x_i)}{(n-1)\sum_{k=1}^{n}\theta(x_k)}; \qquad p_{ij}^{(2)} = \frac{\psi(x_i)\nu_j^{-\frac{1}{2}}}{\sum_{k=1}^{n}\psi(x_k)\sum_{\ell=1}^{n}\nu_\ell^{-\frac{1}{2}} - \sum_{k=1}^{n}\theta(x_k)} \tag{40}$$

and $p_{ii}^{(1)} = p_{ii}^{(2)} = 0$. Set

$$\begin{aligned}
\kappa_{ij} &= 2N\varrho_c(\mu_1 p_{ij}^{(1)} + \mu_2 p_{ij}^{(2)});\\
\mu_1 &= \frac{2(n-1)}{N\varrho_c}\sum_{k=1}^{n}\theta(x_k);\\
\mu_2 &= \frac{2}{N\varrho_c}\sum_{k=1}^{n}\psi(x_k)\sum_{\ell=1}^{n}\nu_\ell^{-\frac{1}{2}} - \frac{2}{N\varrho_c}\sum_{k=1}^{n}\theta(x_k).
\end{aligned} \tag{41}$$

In particular $\widehat{K}(x_i,x_j) = \frac{1}{2}(\kappa_{ij} + \kappa_{ji})$ for $i \neq j$ and $\mu_1 + \mu_2 = 1$. It is important to notice that up to scale the joint probability distributions $p_{ij}^{(k)}$ take a simple product form for $i \neq j$. Computationally this means we may pick particles i and j independently, rather than jointly. If the chosen indices i and j are equal then we reject this choice and try again. This product property of the joint probability distributions leads to a significant improvement in the computationally efficiency of the algorithm.

2.4 The Direct Simulation Algorithm

For $U_s^N \in \mathbb{S}_V^N$ we introduce waiting time parameters associated with sintering and coagulation:

$$\varrho_s(U_s^N) = \sum_{i=1}^{n} \rho_\Lambda(x_i) \qquad \text{and} \qquad \varrho_c(U_s^N) = \frac{1}{2N} \sum_{\substack{i,j=1 \\ i \neq j}}^{n} \widehat{K}(x_i, x_j). \tag{42}$$

It follows that the transition map $\mathcal{P}^N$ is given by

$$\begin{aligned} \mathcal{P}^N(U_s^N) \;\; &= \;\; \frac{\varrho_s}{\varrho_s + \varrho_c} \sum_{i=1}^{n} \frac{\rho_\Lambda(x_i)}{\varrho_s} J_s(U_s^N, x_i) \\ + \frac{\varrho_c}{\varrho_s + \varrho_c} &\sum_{\substack{i,j=1 \\ i \neq j}}^{n} \left\{ \frac{K(x_i, x_j)}{\widehat{K}(x_i, x_j)} J_c(U_s^N, x_i, x_j) + \left(1 - \frac{K(x_i, x_j)}{\widehat{K}(x_i, x_j)}\right) U_s^N \right\} \left(\mu_1 p_{ij}^{(1)} + \mu_2 p_{ij}^{(2)}\right). \end{aligned} \tag{43}$$

In this equation $\varrho_s = \varrho_s(U_s^N)$ and $\varrho_c = \varrho_c(U_s^N)$. We may now use this information to construct an algorithm to generate sample paths for the sintering-coagulation equation:

1. Choose values for the scaling parameter N and the step size parameter Λ. Set the time τ to zero.
2. Find an approximation $p \in \mathbb{S}_V^N$ to the initial particle distribution.
3. Generate a random time $\widehat{\tau}$ which is exponentially distributed with parameter $\varrho_s + \varrho_c$. Increase τ by $\widehat{\tau}$.
4. With probability
$$\frac{\varrho_s}{\varrho_s + \varrho_c} \tag{44}$$
perform a sintering event (step 5), otherwise perform a coagulation event (step 6).
5. Sintering event:
   (a) Pick an index i according to the distribution $\rho_\Lambda(x_i)$.
   (b) Replace p by
   $$p + \frac{1}{N}(\delta_{x_i - \Lambda}^2 - \delta_{x_i}^2). \tag{45}$$
   (c) Return to step 3.

6. Coagulation event:

   (a) With probability μ_1 use $p_{ij}^{(1)}$ to generate indices (step i) otherwise use $p_{ij}^{(2)}$ to generate the indices (step ii).

      i. Pick index i according to the distribution $\theta(x_i)$. and pick j uniformly. If $i = j$ pick another j. Go to step 6(b).

      ii. Pick indices i and j according to the distributions $\psi(x_i)$ and $\nu_j^{-\frac{1}{2}}$ respectively. If $i = j$ reject this choice and perform the step again.

   (b) With probability

$$\frac{K(x_i, x_j)}{\widehat{K}(x_i, x_j)} \tag{46}$$

   replace p by

$$p + \frac{1}{N}(\delta^2_{x_i+x_j} - \delta^2_{x_i} - \delta^2_{x_j}) \tag{47}$$

   otherwise do nothing (a fictitious jump).

   (c) Return to step 3.

This algorithm has the property that at each time supp $p \subseteq \mathcal{R}$. From the definition of ρ_Λ, a particle for which a sintering event would reduce the particle's surface area below the minimum permissible has $\rho_\Lambda(x_i) = 0$, accordingly there is no possibility that such a particle would be chosen to undergo a sintering event. The parameter Λ therefore acts as both a surface area step size and as a parameter that determines when we regard the sintering process for any particular particle as complete.

We have been careful to make sure that for a coagulation event we do not pick equal indices, had we not imposed this requirement the new measure p may not have been non-negative. It is precisely this reason that the new measure μ_τ^N was introduced during the development of the algorithm.

Coagulation events in this algorithm will cause the number of stochastic particles under consideration to decrease with time. In order for the algorithm to cope well with late times it is extremely useful to use the particle doubling method. Since $\mathbb{S}_V^N \subset \mathbb{S}_V^{2N}$ we may set $U_{\tau_0}^{2N} = U_{\tau_0}^N$ if, after a time τ_0, the particle number falls below some preassigned value. We are then able to continue the algorithm with scaling parameter $2N$.

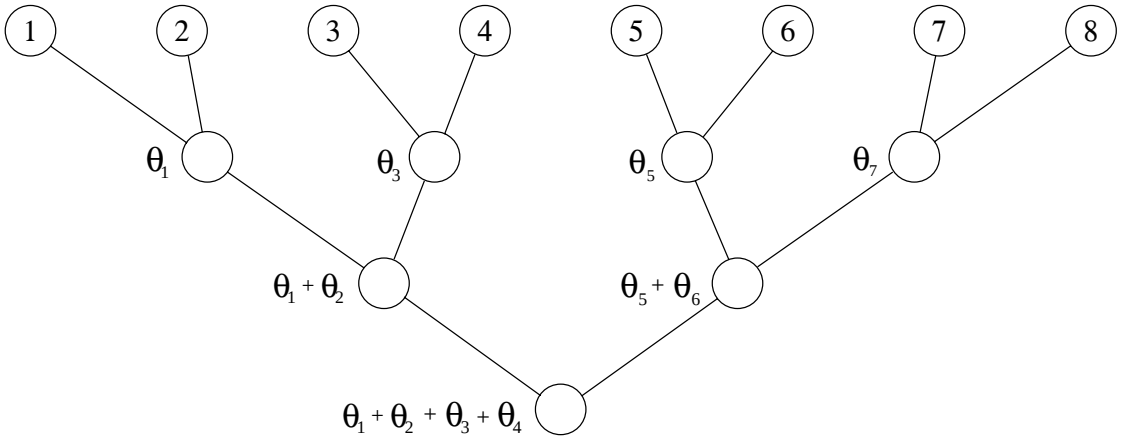


Figure 1: A simple binary tree used to select particles according to a distribution θ

To implement the algorithm just presented it is necessary to choose a particle according to a given distribution. One method of doing this is to use the inversion method implemented using a binary tree. Let the distribution be θ, which need not be normalized. We set up a binary tree as illustrated in Figure 1, each leaf of the tree represents a particle index. In each node of the tree that is not a leaf, we store the cumulative sum of all the contributions to θ obtained by moving to the left of that node. Once the tree is set up we choose a random number Θ, uniformly distributed between 0 and $\sum_i \theta_i$. It is then a simple matter to climb the tree from the root using the information stored in each mode to decide successively which branches one takes until one finds a leaf corresponding to the particle index n such that $\sum_{i=1}^{n-1} \theta_i \leqslant \Theta < \sum_{i=1}^{n} \theta_i$. For a tree of size N it takes $O(\log N)$ operations to climb the tree or to update the relevant quantities in the nodes when a particular θ_i changes due to a coagulation or a sintering event. This method works well in practice, although periodically it is necessary to generate the binary tree from scratch in order to avoid accumulated rounding error from becoming a significant problem.

3 The Mass Flow Algorithm

In this section we show how to implement an alternative algorithm that solves the sintering-coagulation equation. The strategy is to write the original equation in terms of the mass density rather than in terms of the number density. Having done so we can proceed as with the direct simulation algorithm.

Eibeck & Wagner [24], and Wagner [26] describe the new approach which they call the mass flow algorithm. This algorithm has a number of computational advantages over the direct simulation algorithm hitherto exploited, it is also far better at modelling gelling kernels, however this is not an issue for the kernel we are using in this

paper. To cast Eq. (17) into a form suitable for analysis in terms of the mass flow we write $Q(s, \mathrm{d}x) = \nu P(s, \mathrm{d}x)$, so that Q is the measure-valued mass density. We will call $Q \in [0, \infty) \times \mathcal{M}_b(\mathcal{R})$ a measure-valued solution to the sintering-coagulation mass flow equation with initial distribution Q_0 if $Q(0, \cdot) = Q_0$ and

$$\begin{aligned}\int_{\mathcal{R}} \varphi(x) Q(\tau, \mathrm{d}x) &= \int_{\mathcal{R}} \varphi(x) Q(0, \mathrm{d}x) + \int_0^\tau \mathrm{d}s \int_{\mathcal{R}} \{\varphi(x - \Lambda) - \varphi(x)\} \rho_\Lambda(x) Q(s, \mathrm{d}x) \\ &+ \int_0^\tau \mathrm{d}s \int_{\mathcal{R}} \int_{\mathcal{R}} \{\varphi(x + x') - \varphi(x)\} \frac{K(x, x')}{\nu'} Q(s, \mathrm{d}x) Q(s, \mathrm{d}x') \end{aligned} \tag{48}$$

holds for all bounded continuous functions φ defined on $\mathcal{R}$ and for all $\tau > 0$.

Define the jump operator for all $p \in \mathcal{M}_b(\mathcal{R})$ and $x, x' \in \mathcal{R}$ by

$$J_{\mathrm{MFA}}(p, x, x') = p + \frac{1}{N}(\delta^2_{x+x'} - \delta^2_x). \tag{49}$$

For convenience set $\widetilde{K}(x, x') = \widehat{K}(x, x')/\nu'$. Furthermore we will write

$$\phi(x) = \lambda_1 \sigma \nu^{\alpha - 5/2} + \lambda_2 \sigma \nu^{-\frac{3}{2}}; \qquad \omega(x) = \lambda_1 \sigma \nu^{\alpha - 2} + \lambda_2 \sigma \nu^{-1} \tag{50}$$

and

$$\varrho_{\mathrm{MFA}} = \frac{1}{N} \sum_{i,j=1}^n \widetilde{K}(x_i, x_j). \tag{51}$$

Set

$$\varrho_1 = \sum_{k=1}^n \theta(x_k) \sum_{\ell=1}^n \nu_\ell^{-1}; \qquad \varrho_2 = n \sum_{k=1}^n \phi(x_k); \tag{52}$$

$$\varrho_3 = \sum_{k=1}^n \psi(x_k) \sum_{\ell=1}^n \nu_\ell^{-\frac{3}{2}} \qquad \text{and} \qquad \varrho_4 = \sum_{k=1}^n \omega(x_k) \sum_{\ell=1}^n \nu_\ell^{-\frac{1}{2}}. \tag{53}$$

It is a simple matter to verify that

$$\varrho_{\mathrm{MFA}} = \frac{1}{N} \sum_{k=1}^4 \varrho_k. \tag{54}$$

Let us define probabilities by $\mu_k = \dfrac{2\varrho_k}{N \varrho_{\mathrm{MFA}}}$ and define probability distributions $p_{ij}^{(k)}$ by

$$p_{ij}^{(1)} = \frac{\theta(x_i)\nu_j^{-1}}{\varrho_1}, \qquad p_{ij}^{(2)} = \frac{\phi(x_j)}{\varrho_2}, \qquad p_{ij}^{(3)} = \frac{\psi(x_i)\nu_j^{-\frac{3}{2}}}{\varrho_3}, \qquad \text{and} \qquad p_{ij}^{(4)} = \frac{\omega(x_j)\nu_i^{-\frac{1}{2}}}{\varrho_4}. \tag{55}$$

Consequently we have that

$$\frac{\widetilde{K}(x_i, x_j)}{N\varrho_{\mathrm{MFA}}} = \sum_{k=1}^{4} \mu_k p_{ij}^{(k)}. \tag{56}$$

We are now in a position to write down the operator $\mathcal{P}^N$ which determines the transition from one state to the next,

$$\begin{aligned}\mathcal{P}^N(U_s^N) &= \frac{\varrho_s}{\varrho_s + \varrho_{\mathrm{MFA}}} \sum_{i=1}^{n} \frac{\rho_\Lambda(x_i)}{\varrho_s} J_s(U_s^N, x_i) \\ &+ \frac{\varrho_{\mathrm{MFA}}}{\varrho_s + \varrho_{\mathrm{MFA}}} \sum_{i,j=1}^{n} \left\{ \frac{K(x_i, x_j)}{\widehat{K}(x_i, x_j)} J_{\mathrm{MFA}}(U_s^N, x_i, x_j) + \left(1 - \frac{K(x_i, x_j)}{\widehat{K}(x_i, x_j)}\right) U_s^N \right\} \sum_{k=1}^{4} \mu_k p_{ij}^{(k)}\end{aligned} \tag{57}$$

where $\varrho_s = \varrho(U_s^N)$ and $\varrho_{\mathrm{MFA}} = \varrho_{\mathrm{MFA}}(U_s^N)$ in this equation. This is in a form where the algorithm may be easily elucidated.

1. Choose values for the scaling parameter N and the surface area step size parameter Λ. Set the time τ to zero.

2. Find an approximation $p \in \widetilde{\mathbb{S}}_V^N$ to the initial particle distribution where

$$\widetilde{\mathbb{S}}_V^N = \left\{ p \in \mathcal{M}_b(\mathcal{R}) \;\middle|\; p = \frac{1}{N} \sum_{i=1}^{n} \delta_{x_i}^2,\ p(\mathcal{R}) \leqslant V \right\}. \tag{58}$$

3. Generate a random time $\widehat{\tau}$ which is exponentially distributed with parameter $\varrho_s + \varrho_{\mathrm{MFA}}$. Increase τ by $\widehat{\tau}$.

4. With probability

$$\frac{\varrho_s}{\varrho_s + \varrho_{\mathrm{MFA}}} \tag{59}$$

perform a sintering event (step 5), otherwise perform a mass flow coagulation event (step 6).

5. Sintering event:

   (a) Pick an index i according to the distribution $\rho_\Lambda(x_i)$.

   (b) Replace p by

$$p + \frac{1}{N}(\delta_{x_i - \Lambda}^2 - \delta_{x_i}^2). \tag{60}$$

   (c) Return to step 3.

6. Mass flow coagulation event:

   (a) With probabilities μ_k use $p_{ij}^{(k)}$ to generate indices, step (k) below.

   - (1) Pick indices i and j according to the distributions $\theta(x_i)$ and ν_j^{-1} respectively.
   - (2) Pick index j according to the distribution $\phi(x_j)$ and choose i uniformly.
   - (3) Pick indices i and j according to the distributions $\psi(x_i)$ and $\nu_j^{-\frac{3}{2}}$ respectively.
   - (4) Pick indices i and j according the distributions $\nu_i^{-\frac{1}{2}}$ and $\omega(x_j)$ respectively.

   (b) With probability

$$\frac{K(x_i, x_j)}{\widehat{K}(x_i, x_j)} \tag{61}$$

   replace p by

$$p + \frac{1}{N}(\delta^2_{x_i+x_j} - \delta^2_{x_i}) \tag{62}$$

   otherwise do nothing (a fictitious jump).

   (c) Return to step 3.

4 Numerical Results

Both the direct simulation algorithm and the mass flow algorithm provided excellent results. The mass flow algorithm was particularly impressive since it was both fast and accurate. In order to study the convergence of the algorithms we compute functionals of the form

$$\int_{\mathcal{R}} \varphi(x) P(\tau, \mathrm{d}x) \tag{63}$$

which is the limit

$$\lim_{N\to\infty} \frac{1}{N} \sum_{i=1}^{n} \varphi(x_i) \tag{64}$$

in the direct simulation algorithm, and the limit

$$\lim_{N\to\infty} \frac{1}{N} \sum_{i=1}^{n} \frac{\varphi(x_i)}{\nu_i} \tag{65}$$

in the mass flow algorithm. We define (following Eibeck & Wagner [24]) the concentration of particles with volumes within a given range by

$$C_{[a,b]}(\tau) = \int_{\mathcal{R}} \mathbb{1}_{[a,b]}(\nu) P(\tau, \mathrm{d}x) \tag{66}$$

and the volume and area moments

$$M_k(\tau) = \int_{\mathcal{R}} \nu^k P(\tau, \mathrm{d}x); \qquad m_k(\tau) = \int_{\mathcal{R}} \sigma^k P(\tau, \mathrm{d}x) \tag{67}$$

together with the correlation coefficient

$$\rho(\sigma, \nu^{\frac{2}{3}}) = \left(M_{\frac{4}{3}} M_0 - M_{\frac{2}{3}}^2 \right)^{-\frac{1}{2}} \left(m_2 M_0 - m_1^2 \right)^{-\frac{1}{2}} \left[M_0 \int_{\mathcal{R}} \sigma \nu^{\frac{2}{3}} P(\tau, \mathrm{d}x) - m_1 M_{\frac{2}{3}} \right] \tag{68}$$

which measures the extent to which the particles in the ensemble are spherical.

We begin the simulations with a monodisperse initial distribution with unit concentration $P_0 = Q_0 = \delta^2_{(1,1)}$. If we increase the initial particle concentration coagulation events become the dominant process at the start of the simulation, this is because particles are colliding so rapidly that the time scale between collisions is small compared to the characteristic sintering time. At later times the concentration falls and sintering becomes increasingly important compared to the coagulation events. This behaviour is illustrated in Figure 2. The vertical scale is proportional to the particle concentration. It is worth remarking that the region $\mathcal{R}$ is clearly visible. At time $\tau = 0$ we have the initial unit concentration of primary particles. Rapidly coagulation events have occurred and the distribution is spread out predominantly along the line $\sigma = \nu$, corresponding to particle agglomerates where no sintering has taken place. As time progresses, sintering becomes more important, and the particles are distributed with a greater range of volumes and with surface areas that approach that of perfect spheres, this corresponds to the line $\sigma = \nu^{\frac{2}{3}}$. At much later times the primary particles have all undergone coagulation events and the system comprises of a smaller number of large and essentially spherical particles.

It is worth making the point that the particle concentration represented by the graphs in Figure 2 are zero when the volume does not take integral values; in particular the graphs *do not* form smooth surfaces, but rather form a collection of 100 two-dimensional sections supported on $\nu \in \mathbb{N}$. This may be rigorously established from the sintering-coagulation equation as follows. If the initial distribution obeys $\operatorname{supp} P(0, \cdot) \subseteq \{x \in \mathcal{R} \mid \nu \in \mathbb{N}\}$ then taking $\varphi(x) = \nu \mathbb{1}_{\mathbb{R} \setminus \mathbb{N} \cap [0,V]}(\nu)$ and writing $\widetilde{P}(\tau, \cdot) = \mathbb{1}_{\mathbb{R} \setminus \mathbb{N}}(\nu) P(\tau, \cdot)$, we find that Eq. (17) leads to

$$\begin{aligned}
&\int_{\mathcal{R}} \nu \mathbb{1}_{[0,V]}(\nu) \widetilde{P}(\tau, \mathrm{d}x) \\
&\qquad = -\frac{1}{2} \int_0^\tau \mathrm{d}s \int_{\mathcal{R}} \int_{\mathcal{R}} \left\{ \nu \mathbb{1}_{[V-\nu', V]}(\nu) + \nu' \mathbb{1}_{[V-\nu, V]}(\nu') \right\} K(x, x') \widetilde{P}(s, \mathrm{d}x) \widetilde{P}(s, \mathrm{d}x') \\
&\qquad \leqslant 0 \qquad (69)
\end{aligned}$$

for all $V \in \mathbb{R}$. This has the unique solution $\widetilde{P}(\tau, \cdot) = 0$ a.e. for all time τ, as a consequence of the non-negativity of $\widetilde{P}$. It follows that $P(\tau, \cdot) = P(\tau, \cdot)\mathbb{1}_{\mathbb{N}}(\nu)$, i.e., P is supported on $[0, \infty) \times \{x \in \mathcal{R} \mid \nu \in \mathbb{N}\}$ establishing our assertion. We remark that the algorithms presented automatically implement this constraint.

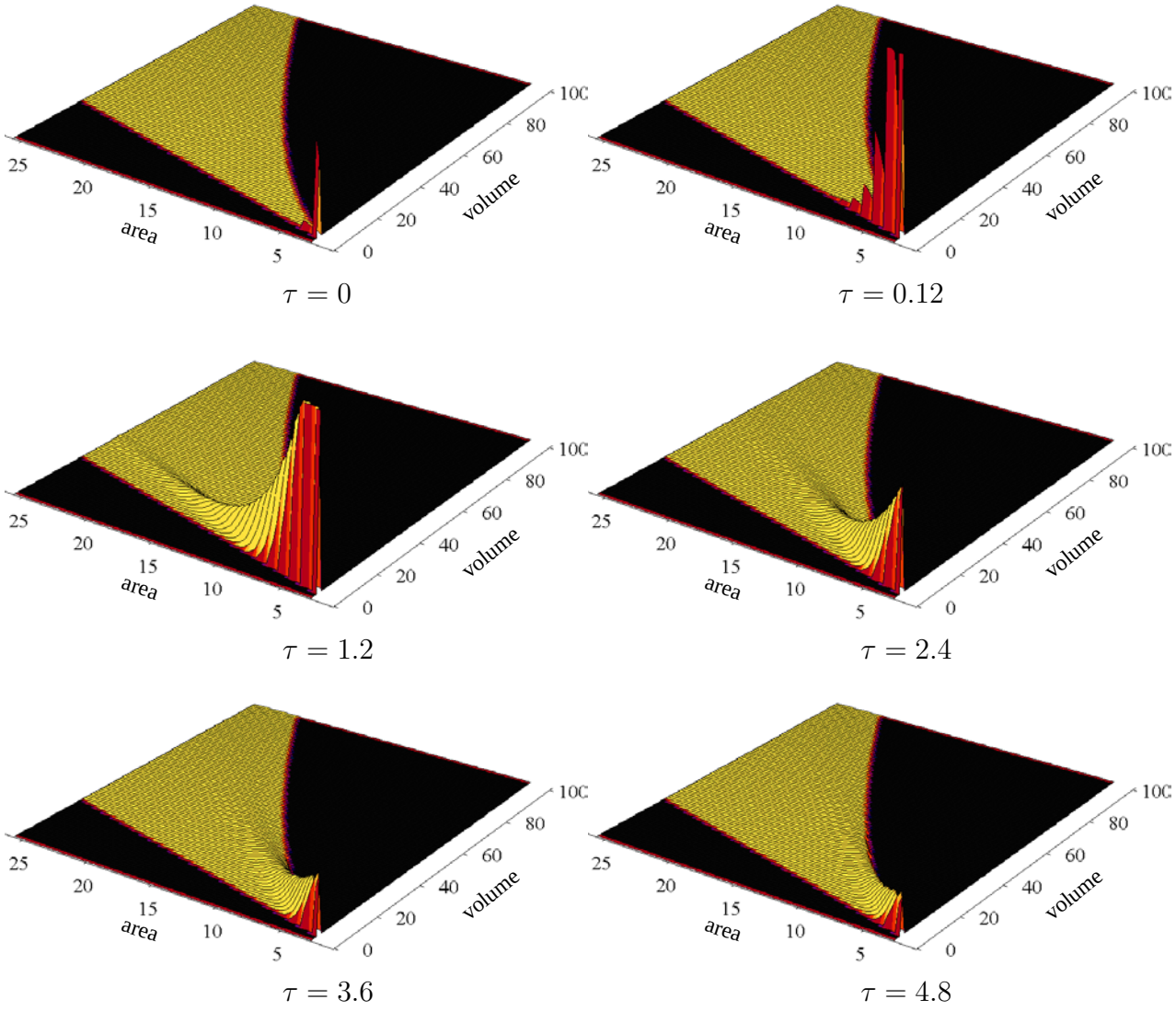


Figure 2: Evolution of the particle volume and area distribution

4.1 Comparison of the Two Algorithms

In this section we compare the direct simulation and mass flow algorithms with parameters $\alpha = 0.8$, $D_F = 2.3$, $\Lambda = 0.01$, initial distribution $P_0 = \delta^2_{(1,1)}$ and a total simulation time of $T = 35$, for a range of values of N. Since at fixed N the direct simulation is faster than the mass flow algorithm, we will be comparing N for the mass flow algorithm with $4.6N$ for the direct simulation, for my implementations, these have very close total running times.

If we assume convergence of the method, the total error one makes when one uses either algorithm is $O(N^{-1})$, this is the best possible since the size of the jumps we make at each event is inversely proportional to N. The extra power of $\nu_i \geqslant 1$ in the denominator of Eq. (65) has the effect of reducing the error for the mass flow algorithm compared to the direct simulation at fixed N. The standard convergence theory, if it can be established, would tell us of the convergence in distribution (of a subsequence) of the sample paths as $N \to \infty$. In order to improve the accuracy it is possible to take a mean over L such sample paths at fixed N, however the statistical error we make in doing so is $O(L^{-\frac{1}{2}})$. Since CPU time is proportional to both L and proportional to $N \log N$, it follows that for a fixed computational time it is better to increase N than to increase L, however other considerations, such as the desire to obtain numerically determined statistical confidence intervals for the results maybe influential in choosing to perform many runs at fixed N.

For the computer implementations of the algorithms described, the mass flow algorithm has a running time of approximately $0.0319N \log N$ seconds per run, and the direct simulation algorithm has a running time of approximately $0.000693N \log N$ seconds per run where particle doubling is implemented if the total number of particles decreases to below 30% of the starting value. The simulations were performed on a stand-alone 1.2 GHz PC with AMD Athlon Processor.

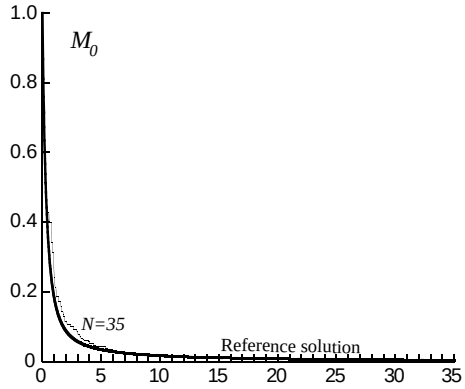
			Relative Error (%)							Error (%)
Alg.	N	Run time	M_0	M_2	M_3	$C_{[100,200]}$	m_1	m_2	m_3	$\rho(\sigma, \nu^{\frac{2}{3}})$
DSA	35	0.14 s	15.861	19.954	37.833	172.495	4.946	47.204	68.635	1.223
MFA	16	0.14 s	9.006	34.720	121.069	72.211	1.852	23.142	60.725	1.511
DSA	1178	4 s	2.647	18.971	65.628	21.797	0.850	14.307	28.872	0.442
MFA	256	4 s	1.690	6.661	15.098	21.088	1.362	8.351	7.919	0.411
DSA	18842	1 m 35 s	0.663	5.788	18.269	8.473	0.291	5.529	7.211	0.209
MFA	4096	1 m 22 s	0.524	1.311	2.802	3.882	0.207	1.480	1.694	0.100
DSA	301466	35 m 26 s	0.079	0.504	1.669	0.869	0.040	0.847	0.957	0.084
MFA	65536	38 m 37 s	0.292	0.207	0.538	0.859	0.140	0.597	0.369	0.031

Table 1: Run times and errors for the algorithms

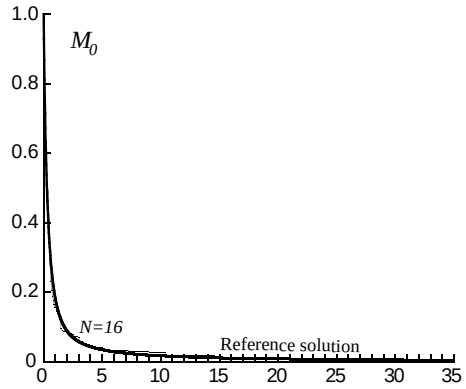
In Table 1 we have defined the absolute error in the computed value of a function $\varphi(\tau)$ using the uniform norm on $[0, T]$, where $T = 35$ is the total simulation time, i.e.,

$$\varphi_{\text{error}} = \|\varphi_N - \varphi\|_\infty = \sup_{t\in[0,T]} |\varphi_N(\tau) - \varphi(\tau)| \tag{70}$$

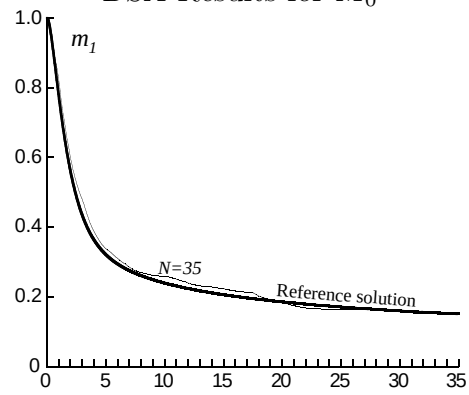
where $\varphi(\tau)$ is the reference solution (obtained by a mass flow algorithm with 2^{22} particles and sintering solved exactly with a run time of 25 h), and φ_N is the estimate derived with N stochastic particles. The table indicates that the mass flow and direct simulations are equally accurate for the zeroth moment, however the mass flow produces significantly more accurate results for the higher volume and area moments for a fixed computational time. The table also suggests convergence of the sample paths generated by each algorithm to the solution of the corresponding sintering-coagulation equation.



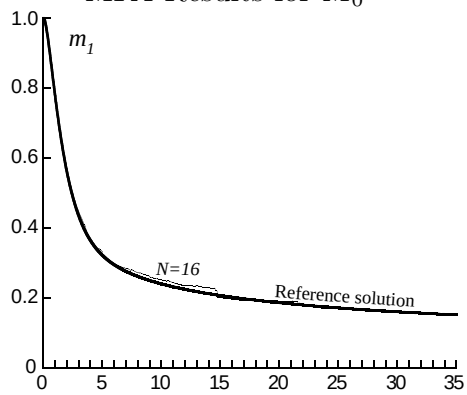
DSA Results for M_0



MFA Results for M_0



DSA Results for m_1

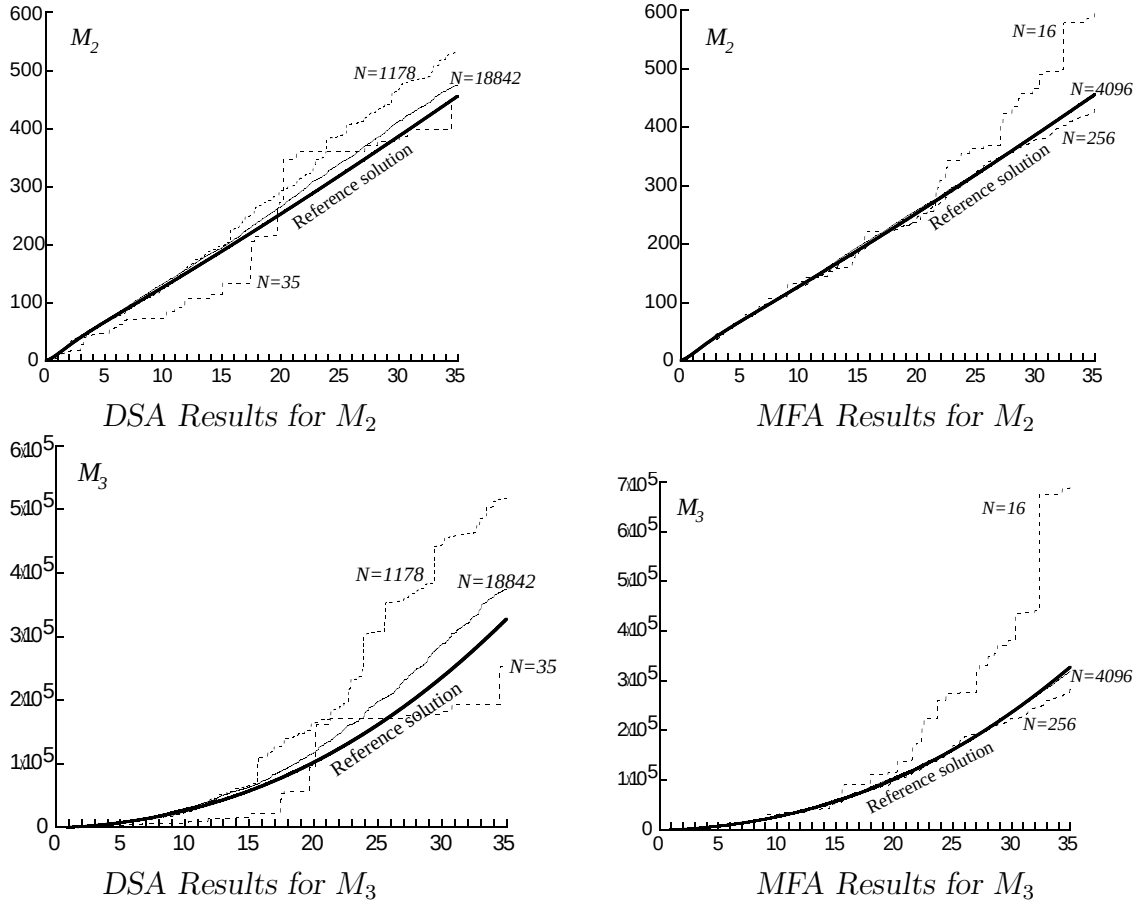


MFA Results for m_1

Figure 3: Graphs of M_0 and m_1

The graphs of M_0 and m_1 illustrate that both algorithms have comparable accuracy for the lowest volume and area moments for a given computation time. In the graphs above only the case with computational time of 0.14 s was plotted, longer simulations

produced results that are graphically indistinguishable from the reference solution.

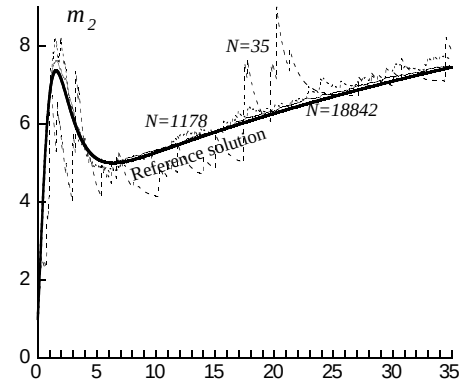


DSA Results for M_2 *MFA Results for M_2*

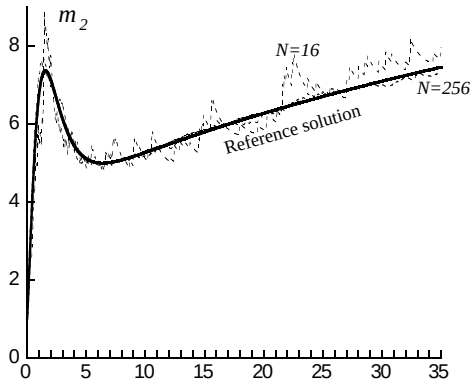
DSA Results for M_3 *MFA Results for M_3*

Figure 4: Graphs of M_2 and M_3

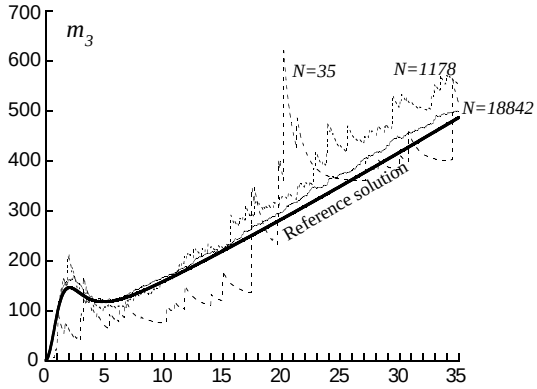
For the higher order volume moments, the mass flow algorithm converges more quickly than the direct simulation. In particular the $N = 4096$ mass flow simulation is almost graphically indistinguishable from the reference solution. The higher surface area moments display the same property as can be seen below.



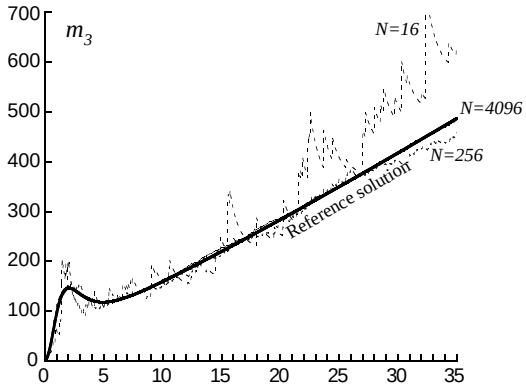
DSA Results for m_2



MFA Results for m_2



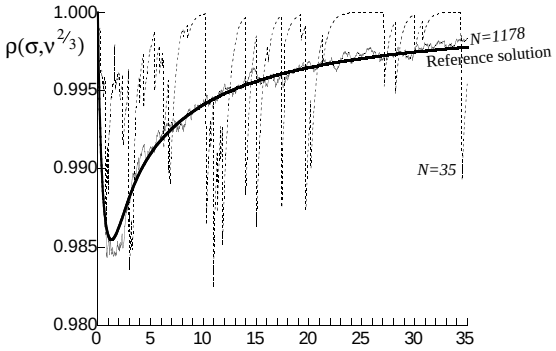
DSA Results for m_3



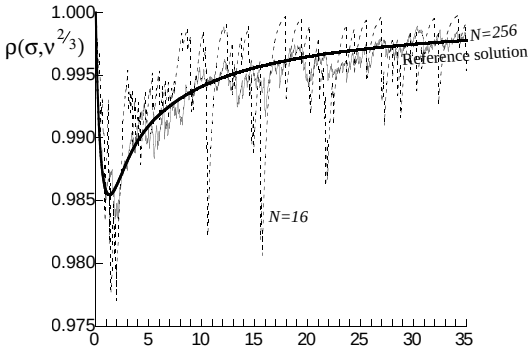
MFA Results for m_3

Figure 5: Graphs of m_2 and m_3

Let us now consider the correlation coefficient $\rho(\sigma, \nu^{\frac{2}{3}})$, the graphs for the two algorithms are shown below.



DSA Results for $\rho(\sigma, \nu^{\frac{2}{3}})$

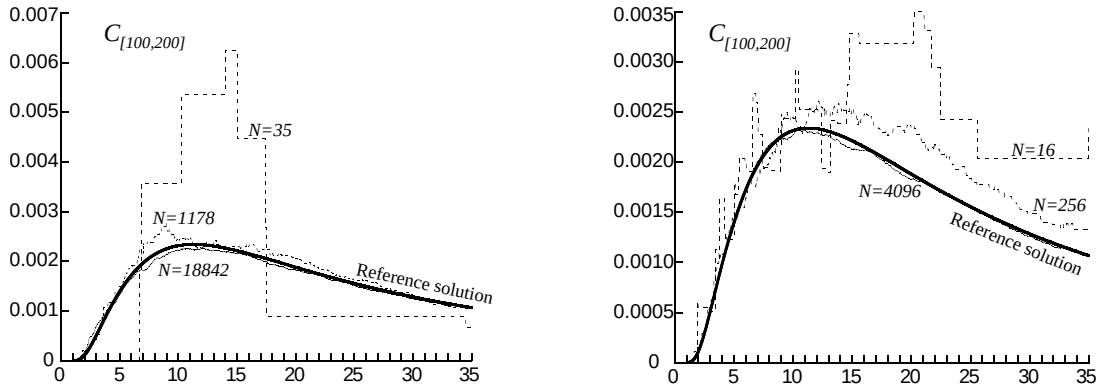


MFA Results for $\rho(\sigma, \nu^{\frac{2}{3}})$

Figure 6: Graphs of the correlation coefficient $\rho(\sigma, \nu^{\frac{2}{3}})$

It should be noted that the vertical axes runs from 0.98 and from 0.975 for the DSA and MFA graphs respectively. Therefore the variations shown in these graphs

are much smaller relative to the size of the solution than might be thought at a cursory glance. Initially we begin the simulation with spherical primary particles, immediately coagulation events dominate leading to non-spherical agglomerates. As the simulation proceeds the sintering events become the dominate process, and the correlation coefficient gradually rises to unity as the remaining particles become increasingly spherical.



DSA Results for $C_{[00,200]}$ *MFA Results for $C_{[100,200]}$*

Figure 7: Graphs of $C_{[100,200]}$

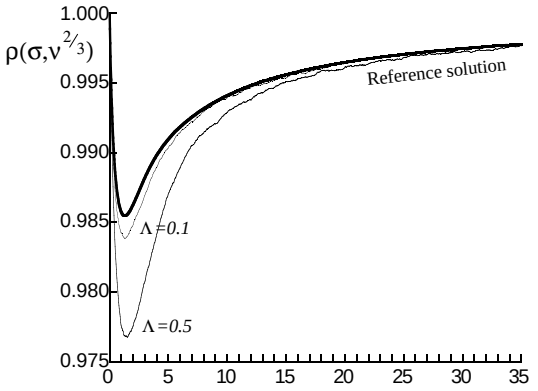
The graphs of $C_{[100,200]}$ depend significantly on the stochastic particle number, which for initial unit concentration equals the scaling parameter N. With only 16 or 35 stochastic particles there is not the resolution to produce accurate answers. For higher N the mass flow and direct simulation algorithms give very similar results, since $C_{[100,200]}$ is a truncated zeroth moment this agrees nicely with our comments for M_0. For larger N, the graphs of $C_{[100,200]}$ and M_2 show surprisingly good agreement with the results presented by Eibeck and Wagner in [24] using the free molecular kernel: $K(x,x') = (\nu^{-\frac{1}{2}} + \nu'^{-\frac{1}{2}})(\nu^{\frac{1}{3}} + \nu'^{\frac{1}{3}})^2$.

4.2 Sensitivity to Variation in the Model Parameter Λ

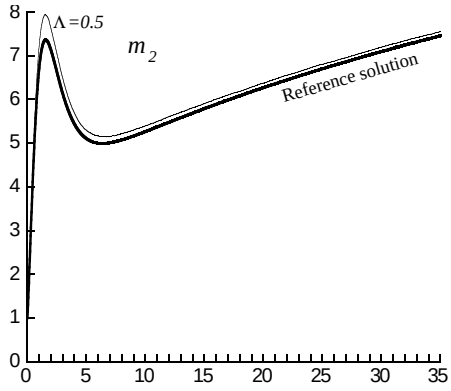
In this section we study the model parameter Λ. To this end we fix $N = 2^{16}$ and run the mass flow simulation at various values of Λ. We have chosen $\Lambda = 0.01$, 0.1 and 0.5. The table below shows the relative error for a single run for each Λ, and the associated run time.

		Relative Error (%)							Error (%)
Λ	Run time	M_0	M_2	M_3	$C_{[100,200]}$	m_1	m_2	m_3	$\rho(\sigma, \nu^{\frac{2}{3}})$
0.01	38 m 37 s	0.292	0.207	0.538	0.859	0.140	0.597	0.369	0.031
0.1	8 m 8 s	0.347	0.857	2.354	0.830	0.135	0.510	1.167	0.167
0.5	5 m 35 s	0.374	1.038	2.024	3.237	2.021	7.835	4.211	0.882

Table 2: Run time and error dependence on Λ



$\rho(\sigma, \nu^{\frac{2}{3}})$ *for* $\Lambda = 0.1,\ 0.5$



m_2 *for* $\Lambda = 0.5$

Figure 8: Graphs of the correlation coefficient $\rho(\sigma, \nu^{\frac{2}{3}})$ *for various* Λ, $N = 2^{16}$

The graphs above show the correlation coefficient and m_2 the second area moment. The graph of the correlation coefficient is consistent with the convergence of the solutions as $\Lambda \to 0$. When $\Lambda = 0.01$, the plot is indistinguishable from the reference solution. The area moments are most susceptible to inaccuracies in the area and therefore to changes in Λ, however only the $\Lambda = 0.5$ simulation produced results visually discernable from the reference solution.

The results presented in the table above suggest convergence in the limit $\Lambda \to 0$ of the solution of the Λ-dependent model given by Eq. (6) to the solution to the model:

$$\begin{aligned}\frac{\partial N_\tau(x)}{\partial \tau} &= \frac{\partial}{\partial \sigma}\left(\left(\sigma - \nu^{\frac{2}{3}}\right) N_\tau(x)\right) \\ &\quad + \frac{1}{2}\int_1^\nu \int_{\nu'^{\frac{2}{3}}}^{\nu'} K(x-x', x') N_\tau(x') N_\tau(x-x')\, \mathrm{d}^2 x' \\ &\quad - \int_1^\infty \int_{\nu'^{\frac{2}{3}}}^{\nu'} K(x, x') N_\tau(x) N_\tau(x')\, \mathrm{d}^2 x'. \end{aligned} \tag{71}$$

A reasonable compromise of accuracy and speed is given by taking $\Lambda = 0.1$ and $N = 4096$ in the mass flow algorithm. This gives good results for all the functionals considered, with the possible exception of $C_{[100,200]}$, in a short computational time.

	Relative Error (%)							Error (%)
Run time	M_0	M_2	M_3	$C_{[100,200]}$	m_1	m_2	m_3	$\rho(\sigma, \nu^{\frac{2}{3}})$
23 s	1.066	1.176	1.467	6.609	0.777	2.424	1.947	0.209

Table 3: Run time and error dependence for the MFA with $N = 4096$ *and* $\Lambda = 0.1$

Given the experimental uncertainty in the measured parameters, and the approximations used in deriving the sintering-coagulation equation, it would be rare that

the moments listed in the table of the solution to the sintering-coagulation equation would be needed to much greater accuracy, though it is elementary to generate more accurate results. As an illustration of the efficiency of these algorithms we remark that the computational time of 23 seconds should be compared with that quoted by Mühlenweg *et al.* [12] of 112 days using the 2D sectional method.

5 Conclusions

In this paper we have studied a model describing the coalescence of a system of particles. This has included an analysis of the evolution of the surface areas of the particles together with their volumes. The methods used provide a significant improvement in efficiency over what has already been achieved by the sectional method, [3, 12] and greater accuracy than less sophisticated Monte Carlo methods, e.g., Rosner & Yu [17].

We have used two methods; the direct simulation and mass flow algorithms to study the system. Each method shows rapid convergence to the solution of the sintering-coagulation equation. There is a marked preference for using the mass flow algorithm when higher volume and area moments are required; the direct simulation algorithm is as good as the mass flow algorithm only for the lowest volume and area moments.

An alternative and in many ways superior method of solving the original sintering-coagulation equation is to make use of the integral curves of the flow defined by the partial differential-integral equation. This has the advantage that there is no need to approximate the derivative with respect to surface area in Eq. (71) by a finite difference, and therefore no need to introduce the model parameter Λ. Effectively one solves a coagulation equation without sintering, but with a time dependent kernel along each integral curve. The essence of the idea is to observe that

$$\left(\frac{\partial N_\tau(x)}{\partial \tau}\right)_{\nu,\sigma} - \left(\frac{\partial}{\partial \sigma}\right)_{\nu,\tau}\left(\left(\sigma - \nu^{\frac{2}{3}}\right) N_\tau(x)\right) = \left(\frac{\partial N_\tau(x)}{\partial \tau}\right)_{\nu,A} - N_\tau(x) \qquad (72)$$

where $\sigma = \nu^{\frac{2}{3}} + (A - \nu^{\frac{2}{3}})e^{-\tau}$. In general this approach requires an understanding of the form of the integral curves, which may not always be analytically tractable or computationally efficient to calculate in other models. In those cases we feel that the approach developed in this paper may provide an efficient numerical algorithm to generate the solution.

References

[1] M. von Smoluchowski, Drei Vortrge über Diffusion, Brownsche Molekularbewegung und Koagulation von Kolloidteilchen, *Phys. Z.,* **17**, 557–571, 585–599, 1916.

[2] W. Koch & S.K. Friedlander, The Effect of Particle Coalescence on the Surface Area of a Coagulating Aerosol, *J. Colloid and Int. Sci.,* **140**(2), 419–427, 1990.

[3] Y. Xiong & S.E. Pratsinis, Particle Formation by Coagulation and Sintering Part I: a Two-Dimensional Solution of the Population Balance Equation, *J. Aerosol Sci.,* **24**, 283–300, 1993.

[4] A. Kazakov & M. Frenklach, Dynamic Modeling of Soot Particle Coagulation and Aggregation: Implementation with the Method of Moments and Application to High-Pressure Laminar Premixed Flames, *Combustion and Flame,* **114**, 484–501, 1998.

[5] D.E. Rosner & J.J. Pyykönen, Bivariate Moment Simulation of Coagulating and Sintering Nanoparticles in Flames, *AIChE Journal,* **48**(3), 476–491, 2002.

[6] B. Schwade & P. Roth, Simulation of Nano-Particle Formation in a Wall-Heated Aerosol Reactor Including Coalescense, *J. Aerosol Sci.,* **34**, 339–357, 2003.

[7] M.J. Hounslow, A Discretized Population Balance for Continuous Systems at Steady State, *AIChE Journal,* **36**(1), 106–116, 1990.

[8] D. Ramkrishna, *Population Balances,* Academic Press, 2000.

[9] D. Lindackers, M.G.D. Strecker, P. Roth, C. Janzen, & S. Pratzinis, Formation and Growth of SiO_2 Particles in Low Pressure $H_2/O_2/Ar$ Flames Doped with SiH_4, *Combust. Sci. and Tech.,* **123**, 287–315, 1997.

[10] Y. Xiong, M.K. Akhtar & S.E. Pratsinis, Particle Formation by Coagulation and Sintering Part II: The Evolution of the Morphology of Aerosol-Made Titania, Silica and Silica-Doped Titania Powders, *J. Aerosol Sci.,* **24**, 301–313, 1993.

[11] S. Tsantilis & S.E. Pratsinis, Evolution of Primary and Aggregate Particle-Size Distributions by Coagulation and Sintering, *AIChE Journal,* **46**(2), 407–415, 2000.

[12] H. Mühlenweg, A. Gutsch, A. Schild & S.E. Pratsinis, Process Simulation of Gas-to-Particle-Sythesis via Population Balances: Investigation of Three Models, *Chem. Eng. Sci.,* **57**, 2305–2322, 2002.

[13] S. Tsantilis, H.K. Kammler, & S.E. Pratsinis, Population Balance Modelling of Flame Synthesis of Titania Nano-Particles, *Chem. Eng. Sci.,* **57**, 2139–2156, 2002.

[14] M. Zurita-Gotor & D.E. Rosner, Effective Diameters for Collisions of Fractal-like Aggregates: Recommendations for Improved Aerosol Coagulation Frequency Predictions, *J. Colloid and Int. Sci.,* **255**, 10–26, 2002.

[15] D.T. Gillespie, The Stochastic Coalescence Model for Cloud Droplet Growth, *J. Atmosph. Sci.,* **29**, 1496–1510, 1972.

[16] P. Tandon & D.E. Rosner, Monte Carlo Simulation of Particle Aggregation and Simultaneous Restructuring, *J. Colloid and Int. Sci.,* **213**, 273–286, 1999.

[17] D.E. Rosner & S. Yu, MC Simulation of Aerosol Aggregation and Simultaneous Spheroidization, *AIChE Journal,* **47**, 545–561, 2001.

[18] F.E. Kruis, A. Maisels, & H. Fissan, Direct Simulation Monte Carlo Method for Particle Coagulation and Aggregation, *AIChE Journal,* **46**, 1735–1742, 2000.

[19] J.R. Norris, Smoluchowski's Coagulation Equation: Uniqueness, Non-uniqueness and a Hydrodynamic Limit for the Stochastic Coalescent, *Ann. Appl. Probab.,* **9**(1), 78–109, 1999.

[20] J.R. Norris, Cluster Coagulation, *Comm. Math. Phys.,* **209**(2), 407–435, 2000.

[21] I. Jeon, Existence of Gelling Solutions for Coagulation-Fragmentation Equations, *Comm. Math. Phys.,* **194**(3), 541–567, 1998.

[22] A. Eibeck & W. Wagner, Approximative Solution of the Coagulation-Fragmentation Equation by Stochastic Particle Systems, *Stochastic. Anal. Appl.,* **18**(6), 921–948, 2000.

[23] A. Eibeck & W. Wagner, An Efficient Stochastic Algorithm for Studying Coagulation Dynamics and Gelation Phenomena, *Siam J, Sci. Comput.,* **22**(3), 802–821, 2000.

[24] A. Eibeck & W. Wagner, Stochastic Particle Approximations for Smoluchowski's Coagulation Equation, *Preprint 585, Weierstraß-Institut für Angewandte Analysis und Stochastik, Berlin,* 2000.

[25] A. Eibeck & W. Wagner, Stochastic Interacting Particle Systems and Nonlinear Kinetic Equations, *Preprint 732, Weierstraß-Institut für Angewandte Analysis und Stochastik, Berlin,* 2002.

[26] W. Wagner, Stochastic, Analytic and Numerical Aspects of Coagulation Processes, *Preprint 697, Weierstraß-Institut für Angewandte Analysis und Stochastik, Berlin,* 2001.

[27] M. Goodson & M. Kraft, An Efficient Stochastic Algorithm For Simulating Nano-particle Dynamics, *J. Comput. Phys.,* **183**(1), 210–232, 2002.

[28] M. Balthasar & M. Kraft, A Stochastic Approach to Solve the Particle Size Distribution Function of Soot Particles in Laminar Premixed Flames, *Combustion and Flame,* 2003, in press.

[29] D. Grosschmidt, H. Bockhorn, M. Goodson, & M. Kraft, Two Approachs to the Simulation of Silica Particle Synthesis, *Proceedings of the Combustion Institute,* 2003, in press.