Weierstraß-Institut
für Angewandte Analysis und Stochastik

im Forschungsverbund Berlin e.V.

Preprint ISSN 0946 – 8633

Stochastic particle methods for Smoluchowski coagulation equation: variance reduction and error estimations*

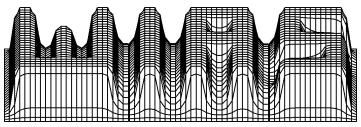
Anastasia Kolodko[1,2] and Karl Sabelfeld[1,2]

submitted: April 26, 2003

[1] Weierstrass Institute for Applied Analysis and Stochastics
Mohrenstra"se 39
D – 10117 Berlin
Germany
E-Mail: sabelfeld@wias-berlin.de

[2] Institute of Computational Mathematics and Mathematical Geophysics
Russian Acad. Sci.
Lavrentieva str., 6
630090 Novosibirsk, Russia
E-Mail: kolodko@wias-berlin.de

No. 842
Berlin 2003



1991 *Mathematics Subject Classification.* 65C05, 76N20.

Key words and phrases. Stochastic particle methods, Smoluchowski equation, variance reduction, coagulation-fragmentation process.

* Support by the Russian Fund of Fundamental Studies under Grant N 03-01-00914, the European Grant INTAS99-1501, NATO Linkage Grant N 978912, the Alexander von Humboldt Foundation, and DFG grant N 436RUS17/84/02 is kindly acknowledged..

Edited by
Weierstraß-Institut für Angewandte Analysis und Stochastik (WIAS)
Mohrenstraße 39
D — 10117 Berlin
Germany

Fax: + 49 30 2044975
E-Mail: preprint@wias-berlin.de
World Wide Web: http://www.wias-berlin.de/

Abstract

Stochastic particle methods for the coagulation-fragmentation Smoluchowski equation are developed and a general variance reduction technique is suggested. This method generalizes the mass-flow approach due to H. Babovsky, and has in focus the desired band of the size spectrum. Estimations of the variance and bias of the method are derived. A comparative cost and variance analysis is made for the known stochastic methods. An applied problem of coagulation-evaporation dynamics in free molecule regime is solved.

1 Introduction

The coagulation-fragmentation processes in spatially homogeneous flows are governed by the Smoluchowski equation (e.g., see [19]):

$$\frac{\partial u_l(t)}{\partial t} = \frac{1}{2}\sum_{i+j=l}\left(K_{ij}u_iu_j - F_{ij}u_l\right) - \sum_{i\geq 1}\left(K_{li}u_lu_i - F_{li}u_{l+i}\right); \qquad l \geq 1. \tag{1}$$

Here u_l is the number density of the clusters, containing l structural units ($\{l\}$-cluster); K_{ij} is the coagulation coefficient, characterizing the collision frequencies between the $\{i\}$- and $\{j\}$-clusters; F_{ij} is the fragmentation coefficient, characterizing the probability of fragmentation of $\{i+j\}$-cluster into $\{i\}$- and $\{j\}$-clusters.

We will use the notation $u(t)$ for the vector $(u_l(t))_{l\geq 1}$.

The existence and uniqueness of a mass-preserving solution to (1) have been established for rather general assumptions about the coagulation and fragmentation coefficients (e.g., see [3]). In our study these assumptions are accepted. In fact, we will sometimes use even stronger assumptions.

The Smoluchowski equation is rather difficult to solve by deterministic numerical methods. The main point is the high dimension. On the other hand, it has clear physical and probability structure. That is why the stochastic algorithms based on statistical simulation of the coagulation-fragmentation processes play a crucial role in the numerical analysis of the coagulation equations.

Let us descibe the general scheme of stochastic particle algorithm.

A system of N particles is considered, whose initial size distribution is given. Its state at the time instant t is defined by the vector

$$\{l_1(t), \ldots, l_L(t)\},$$

where l_i is the size of the i-th particle, or $\{l_i\}$-cluster. The change of the state happens in discrete time steps. At each time step the particles collide according to some probabilities,

which may born particles of larger size, or (and) disintegrate large particles into smaller particles. To the time t, our system consists of L particles with a certain size distribution. A random estimator of the solution to (1) is defined as follows:

$$U_l(t;\, N) = \frac{1}{N} \sum_{i=1}^{L} \delta_{l,l_i(t)}, \quad l \geq 1.$$

For simplicity, we will use the notation $U^N(t)$ for the vector $(U_l(t;\, N))_{l\geq 1}$, and $\|U^N\|$ is $\|\,\|$-norm defined as $\|U^N\| = \sum_{l\geq 1} |U_l(t;\, N)|$.

Stochastic algorithms are all based on the statement that under certain assumptions, the ensemble average of the random process $U^N(t)$ should converge, as $N \to \infty$, to the solution of Smoluchowski equation (1).

A simple way to model the coagulation is a direct simulation of the coagulation process, called also a Marcus-Lushnikov process: at each time step the $\{l_i\}$- and $\{l_j\}$-clusters are sampled according to the rate K_{l_i,l_j}, which results in a new particle of size $l_i + l_j$. These models (for pure coagulation case) were introduced in [7, 13, 14]. These algorithms are analogous to Bird algorithms for Boltzmann equation. Later, some modifications of these algorithms have been developed, improving their efficiency, such as the "method of majorant frequencies" (see [9, 16],) and its generalization – the "method of fictitious jumps" (see [5]). The convergence of the Marcus-Lushnikov process to the solution to (1) has been proved under some broad assumptions about the coagulation and fragmentation coefficients (see [5]).

Some stochastic algorithms involve an additional parameter, a multiplier of the time step, thus allowing collisions between several pairs of particles during one time step. These algorithms are analogous to the Nanbu algorithm for the Boltzmann equation [15]. Their description are presented in [12, 16]. The convergence of the Nanbu type method to Smoluchowski equation in pure coagulation case have been studied by us in [11, 17]. There we have shown, that the random estimator converges in probability, as $N \to \infty$, to the discretized analogue of (1), generated by the time step parameter α.

Convergence is one of the most important issues in the theory of stochastic numerical algorithms, along with the error estimation. The following questions always arise, when dealing with stochastic algorithms:

- Which values of parameters (here: the number of particles N, the time step parameter α) should we take to achieve the required level of accuracy?
- How does the value of the error depend on the parameters of the algorithm? How should we change the values of the parameters to get, say, two times smaller error?

Though the calculations show, that the error is inverse proportional to the number of particles N (see [10, 11]), theoretically this problem is not solved yet.

When tackling the error of stochastic algorithms for the calculation of some function $h(u)$ of the solution to Smoluchowski equation, we will consider two parts of the error. First, the bias of the estimator

$$B^N = ||\mathbb{E}[h(U^N)] - h(u)||,$$

provided N is bounded. Second, the statistical error

$$\mathrm{St}^{N,M} = \sqrt{\frac{||\mathrm{Var}[h(U^N)]||}{M}},$$

where M is the number of samples (or, simply saying, the number of independent algorithm realizations).

It is clear, that the statistical error can be decreased by two ways: (1) diminishing the variance of the estimator, and (2) enlarging the statistics, i.e., increasing the number of realizations of the algorithm. A fundamental characteristic of an algorithm is its efficiency Eff for calculating the function $h(u)$. To define it, note, that the number M_ε of the samples, needed to achieve the desired value of the statistical error

$$\mathrm{St}^{N,M} = \varepsilon$$

is equal to

$$M_\varepsilon = \frac{||\mathrm{Var}[h(U^N)]||}{\varepsilon^2}$$

Then, the computational cost of the algorithm is

$$C_\varepsilon = \tau M_\varepsilon = \frac{||\mathrm{Var}[h(U^N)]||\tau}{\varepsilon^2},$$

where τ is the time necessary to run 1 sample. The efficiency of the algorithm is then defined as

$$\mathrm{Eff} = \frac{1}{C_\varepsilon}\ . \tag{2}$$

In this paper we deal with both parts of the error, for the Bird type algorithm. In Section 2 we present the analytical estimation of the bias. In Section 3 we discuss the methods of the reduction of the variance and then, present a comparative analysis of the efficiencies of different algorithms in Section 4.

One of the main reasons of the large variance is that the number of particles in the model system decreases with time. To overcome it, a special procedure have been developed in [16]. Namely, the coagulation processes were simulated independently in different systems, which were mixed after the number of particles became small, providing new full systems. Another effective way to preserve the number of particles in a model system is simply to double the number of certain clusters, if this number becomes two times smaller ([12, 11, 16]).

These procedures, however, cannot get over another drawback of the stochastic algorithms: the relatively small number of large particles, causing the relatively high variance of the concentration of large particles. The way to reduce this variance have been proposed in [2] and then studied in [4]. The main idea is to turn from the equation (1), governing the particle concentrations, to the equation, governing particle masses (mass-flow equation). Now, the mass-preserving property of the model system provides the constant number of particles. The mass-flow equation is obtained from the original equation via multiplication of the particle concentration by the particle size. This equation governs then the particle mass, and his coagulation kernel changes the dynamics of the coagulation process in such a way that the number of large particles is decreasing much slower than in the process described by the original Smoluchowski equation.

In the present paper we suggest a generalization of the mass-flow approach, namely, a Weight Algorithm, the main idea of which is to turn to the Smoluchowski equation written for the product $u_l(t)\, w(l)$ where $w(l)$ is a weight function ($w(l) = l$ is the particular case of mass). We study the efficiency of the algorithms discussed according to our definition (2).

2 Bias of the Bird Type Estimator

We will deal with the pure coagulation equation

$$\frac{\partial u_l(t)}{\partial t} = \frac{1}{2}\sum_{i+j=l} K_{ij}u_iu_j - \sum_{i\geq 1} K_{li}u_lu_i; \quad l \geq 1 \tag{3}$$

with monodisperse initial conditions

$$u_1(0) = 1; \quad u_l(0) = 0,\ l \geq 2. \tag{4}$$

We assume that the coagulation coefficients are finite, i.e. there exists $k_{\max} < \infty$ such that $\max\limits_{i,j} K_{ij} \leq k_{\max}$. Then, as we have already noticed in the Introduction, there exists a unique solution to problem (3)-(4) conserving its mass.

Let $U^N(t)$ be the random estimator, constructed by the direct simulation, or Bird type method, e.g., see [16]. Then, let us denote by $cov^N(t)$ the following vector, related to the estimator: $cov^N(t) = \left(\sum\limits_{j\geq 1} \mathrm{Cov}\left[U_i^N(t), U_j^N(t)\right]\right)_{i\geq 1}$.

The following statement provides the estimation of the bias of the Bird type method.

Theorem 2.1.
For each N and $t \in [0;\, T]$ the following inequality is true:

$$||\mathbb{E}[U^N(t)] - u(t)|| \leq \left(\frac{3}{4N} + \sup_{\tau\leq t} ||cov^N(\tau)||\right) \cdot \frac{\exp(4k_{\max}t) - 1}{2}\ .$$

Before proving the theorem, we present two simple lemmas.

Lemma 2.1.
Let $\mathcal{K}(x) = (\mathcal{K}_l(x))_{l\geq 1}$ be the following operator on X:

$$\mathcal{K}_l(x) = \frac{1}{2}\sum_{i+j=l} K_{ij}x_ix_j - x_l\sum_{i\geq 1} K_{li}x_i.$$

The following inequalities are true:

$$1) \quad ||\mathcal{K}(x) - \mathcal{K}(y)|| \leq 4k_{\max}||x - y|| \quad \textit{for each } x, y \in X \tag{5}$$

$$2) \quad ||\,\mathbb{E}\left[\mathcal{K}\left(U^N(t)\right)\right] - \mathcal{K}\left(\mathbb{E}\left[U^N(t)\right]\right)\,|| \leq 2k_{\max}||cov^N(t)||. \tag{6}$$

Lemma 2.2.
Let $\theta(t)$ be a function, continuous on $[0;\, T]$ and suppose that for each $t \in [0;\, T]$ the following inequality is true: $\theta(t) \leq \int\limits_0^t (\alpha\theta(\tau) + \beta)\, d\tau$, where α and β are constants.

Then, for each $t \in [0;\, T]$ the following inequality is true:

$$\theta(t) \leq \frac{\beta}{\alpha}\left(\exp(\alpha t) - 1\right). \tag{7}$$

Proof of Statement.
Let us denote $p_N(l_1,\dots,l_N;\,t)$ the probability density function (pdf) of the states of the model system in Bird's method, and let

$$p_N^{(2)}(l_i,l_j;\,t)=\sum_{l_1\geq 0}\dots\sum_{l_{i-1}\geq 0}\sum_{l_{i+1}\geq 0}\dots\sum_{l_{j-1}\geq 0}\sum_{l_{j+1}\geq 0}\dots\sum_{l_N\geq 0}p_N(l_1,\dots,l_N;\,t);$$

$$p_N^{(1)}(l_i;\,t)=\sum_{l_1\geq 0}\dots\sum_{l_{i-1}\geq 0}\sum_{l_{i+1}\geq 0}\dots\sum_{l_N\geq 0}p_N(l_1,\dots,l_N;\,t).$$

It has been shown in [16] that the following equation is true for $p_N^{(1)}(l;\,t)$:

$$\frac{\partial}{\partial t}p_N^{(1)}(l;\,t)=\frac{N-1}{N}\left(\frac{1}{2}\sum_{i+j=l}K_{ij}p_N^{(2)}(i,j;\,t)-\sum_{i\geq 1}K_{il}p_N^{(2)}(l,i;\,t)\right);\quad l\geq 1. \tag{8}$$

Note, that

$$\mathbb{E}\left[U_m^N(t)U_n^N(t)\right]=\frac{1}{N^2}\sum_{l_1\geq 0}\dots\sum_{l_N\geq 0}\left(\sum_{i=1}^{N}\sum_{j=1}^{N}\delta_{ml_i}\delta_{nl_j}p_N(l_1,\dots,l_N;\,t)\right)=$$

$$=\frac{1}{N^2}\sum_{l_1\geq 0}\dots\sum_{l_N\geq 0}\left(\left(\sum_{i=1}^{N}\sum_{j=1,j\neq i}^{N}+\sum_{i=1}^{N}\sum_{j=i}\right)\delta_{ml_i}\delta_{nl_j}p_N(l_1,\dots,l_N;\,t)\right)=\frac{1}{N^2}\left(S_1+S_2\right) \tag{9}$$

where

$$S_1=\sum_{i=1}^{N}\sum_{j=1,j\neq i}^{N}\sum_{l_i\geq 0}\sum_{l_j\geq 0}\delta_{ml_i}\delta_{nl_j}p_N^{(2)}(l_i,l_j;\,t)=N(N-1)p_N^{(2)}(m,n;\,t);$$

$$S_2=\sum_{i=1}^{N}\sum_{l_i\geq 0}\delta_{ml_i}\delta_{nl_i}p_N^{(1)}(l_i;\,t)=N\delta_{mn}p_N^{(1)}(m;\,t).$$

Finally,

$$\mathbb{E}\left[U_m^N(t)U_n^N(t)\right]=\frac{N-1}{N}p_N^{(2)}(m,n;\,t)+\frac{1}{N}\delta_{mn}p_N^{(1)}(m;\,t). \tag{10}$$

Analogously, $\mathbb{E}\left[U_l^N(t))\right]=p_N^{(1)}(l;\,t)$.

Then, (10) yields

$$\frac{N-1}{N}p_N^{(2)}(i,j;\,t)=\mathbb{E}\left[U_i^N(t)U_j^N(t)\right]-\frac{1}{N}\delta_{mn}p_N^{(1)}(m;\,t),$$

and equations (8) can be written as follows:

$$\frac{\partial}{\partial t}\mathbb{E}[U_l^N(t)]=\frac{1}{2}\sum_{i+j=l}K_{ij}\mathbb{E}[U_i^N]\mathbb{E}[U_j^N]-\mathbb{E}[U_l^N]\sum_{i\geq 1}K_{il}\mathbb{E}[U_i^N]+$$

$$+\frac{1}{2}\sum_{i+j=l}K_{ij}\mathrm{Cov}[U_i^N,U_j^N]-\sum_{i\geq 1}K_{il}\mathrm{Cov}[U_i^N,U_l^N]-\frac{1}{N}\left(R_{1l}(t)+R_{2l}(t)\right),\quad l\geq 1 \tag{11}$$

where

$$R_{1l}(t)=\frac{1}{2}\sum_{i+j=l}K_{ij}\delta_{ij}-\sum_{i\geq 1}K_{il}\mathbb{E}[U_i^NU_l^N]\,,$$

$$R_{2l}(t) = \frac{1}{2}\sum_{i+j=l} K_{ij}\delta_{ij}p_N^{(1)}(i;\,t) - \sum_{i\geq 1} K_{il}\delta_{il}p_N^{(1)}(i;\,t)$$
$$= \frac{1}{2}I(odd(l))K_{\frac{l}{2}\frac{l}{2}}p_N^{(1)}\left(\frac{l}{2};\,t\right) - K_{ll}p_N^{(1)}(l;\,t).$$

Note that

$$||R_1(t)|| \leq 2\sum_{i\geq 1}\sum_{j\geq 1} K_{ij}\mathbb{E}[U_i^N U_j^N] \leq 2k_{\max};$$

$$||R_2(t)|| \leq \frac{1}{2}k_{\max}\sum_{l\geq 1} I(odd(l))p_N^{(1)}\left(\frac{l}{2};\,t\right) + k_{\max}\sum_{l\geq 1} p_N^{(1)}(l;\,t) \leq \frac{3k_{\max}}{2}. \tag{12}$$

Using (3), (11), (12) and inequalities of lemma 2.1, we get

$$\frac{\partial}{\partial t}||\mathbb{E}[U^N(t)] - u(t)|| \leq 4k_{\max}||\mathbb{E}[U^N(t)] - u(t)|| + 2k_{\max}||cov^N(t)|| + \frac{3k_{\max}}{2N},$$

and according to the choice of $U(0)$,

$$||\mathbb{E}[U^N(t)] - u(t)|| \leq \int_0^t \left(4k_{\max}||\mathbb{E}[U^N(\tau)] - u(\tau)|| + 2k_{\max}||cov^N(\tau)|| + \frac{3k_{\max}}{2N}\right)d\tau.$$

Then, lemma 2.2 yields the inequality, and Theorem 2.1 is proved.

The statement shows, that the vector $cov^N(t) = (\sum_{j\geq 1}\mathrm{Cov}[U_i^N(t), U_j^N(t)])_{i\geq 1}$ can be used to estimate the value of the bias of the Bird type method. Though this value cannot be estimated analytically, the calculations show, that it is inverse to the number of particles in the model system N. In Table 2.1 we show the values of several components of the vector $cov^N(t)$ (their maximal values on $t \in [0;\,T]$) for different values of N for $K_{ij} = 1$ and $T = 20$.

Table 2.1. The value of $\sup\limits_{t\in[0;T]} cov_l^N(t)$ for $K_{ij} = 1$ and $T = 20$.

N	$l = 1$	$l = 2$	$l = 4$	$l = 8$	samples
128	1.62e-3	2.58e-4	1.21e-4	2.25e-5	240000
256	8.11e-4	1.32e-4	6.05e-5	1.12e-5	120000
512	4.06e-4	6.80e-5	3.09e-5	5.47e-6	60000
1024	2.06e-4	3.45e-5	1.52e-5	2.89e-6	30000

Calculations of the value of $\sup\limits_{t\in[0;\,T]} ||\mathbb{E}[U^N(t)] - u(t)||$ (see Table 2.2) show, that it decreases proportionally to N^{-1} as N increases. Therefore, in view of the estimation given in Theorem 2.1, we conclude that the norm of the vector $cov^N(t)$ behaves like N^{-1}, as N increases.

Table 2.2. The value of $\sup_{t\in[0;T]} ||\mathbb{E}[U^N(t)] - u(t)||$ for $K_{ij} = 1$ and $T = 20$.

N	$\sup_{t\in[0;T]} ||\mathbb{E}[U^N(t)] - u(t)||$	samples
128	9.17e-3±0.21e-3	240000
256	4.59e-3±0.21e-3	120000
512	2.28e-3±0.21e-3	60000
1024	1.11e-3±0.21e-3	30000

3 Variance Reduction Algorithms

3.1 Weight Equation

As we have already mentioned in the Introduction, the decrease of the number of particles in a model system in stochastic algorithms is the main source of the variance increase. To overcome it, H. Babovsky [2] has suggested to transform the equation (1) (governing the particle concentration) to the mass-flow equation

$$\frac{\partial g_l(t)}{\partial t} = \sum_{i=1}^{l-1} \frac{1}{i} K_{i(l-i)} g_i g_{l-i} - g_l \sum_{i\geq 1} \frac{1}{i} K_{li} g_i,\ l \geq 1; \quad \text{where} \quad g_l(t) = l u_l(t).$$

Stochastic algorithms for this equation (for their detailed description, see [2, 4]) keep the number of particles fixed during the whole time. This approach has also another nice property: the number of large particles decreases slower than in the traditional Bird's algorithm. One may expect that these features should drastically increase the variance.

However, the distribution of small and large particles in mass-flow systems is still far from being uniform. In fact, even in the case of constant coagulation coefficients $K_{ij} = 1$ the decrease of particle concentration with the growth of size l (for any fixed time t) is exponential:

$$u_l(t) = (1 + 0.5t)^{-2} \left(1 - \frac{1}{1 + 0.5t}\right)^{l-1}.$$

Note that, for the unbounded coagulation kernels the relative number of large particles is even smaller.

Therefore, it is reasonable to introduce a weight function $w(l)$ increasing with the growth of l, and to transform the original Smoluchowski equation (1) to a "weight equation" for the new variable, the weighted concentration $z_l(t) = w(l)u_l(t)$. This weight equation reads obviously

$$\frac{\partial z_l(t)}{\partial t} = \frac{1}{2} \sum_{i+j=l} \hat{K}_{ij} z_i z_j - z_l \sum_{i\geq 1} \hat{K}_{li} \frac{w(l)}{w(i+l)} z_i, \quad l \geq 1; \qquad \hat{K}_{ij} = K_{ij} \frac{w(i+j)}{w(i)w(j)}. \quad (13)$$

To solve this equation, we develop a generalization of the method of fictitious jumps (MFJ) (see [5] for (1)). Together with this algorithm, we construct a generalization of the method of majorant frequencies (MMF) (its description for (1) can be found in [16]). Though the latter algorithm can be treated as the particular case of the former, it is very important itself, as comes from the remarks below.

3.2 Description of the Algorithms

Let us consider a system of N particles. Each particle i is characterized by its size l_i, where l_i is an integer number.

Suppose, that we know a function h such that $\hat{K}_{ij} \le h(i)h(j)$ for each i, j.

The initial size distribution of the system is constructed arbitrarily, but in such a way that it approximates the initial conditions for (13):

$$\frac{\sum\limits_{p=1}^{N} \delta_{i,l_p}}{N} \approx z_i(0), \quad i \ge 1.$$

The changes of the states of the system happen in discrete time steps. At each time step a pair of particles i and j is chosen, which coagulate according to some probability, that is a particle of size $l_i + l_j$ is born.

Let us describe the evolution process in more details, both for MFJ and MMF methods. Assume the state of the system at the time instant t_k is $(l_1, \dots, l_L)$. Then, its state at the time instant t_{k+1} is defined as follows.

1. Choose a random time step τ according to the distribution

   $$p_\tau(x) = \lambda \exp(-\lambda x),$$

   where $\lambda = \dfrac{2N}{\left(\sum\limits_{i=1}^{L} h(l_i)\right)^2 W_{\max}}$ for MFJ

   and

   $$\lambda = \frac{2N}{L(L-1) \max\limits_{1 \le i,j \le L} \hat{K}_{l_i l_j} W_{\max}} \quad \text{for MMF}, \tag{14}$$

   where $W_{\max} = \max\limits_{1 \le i,j \le L} \dfrac{w_{l_i + l_j}}{w_{l_i} + w_{l_j}}$.

   Set $t_{k+1} = t_k + \tau$.

2. Choose the reacting particles i and $j \ne i$ according to the distribution

   $$h(l_i) / \sum_{j=1}^{L} h(l_j), \quad i = 1, \dots, L$$

   in the MFJ method, and uniformly distributed in MMF method.

3. Check, whether the interaction takes place between l_i and l_j with probability $\dfrac{\hat{K}_{l_i l_j}}{h(l_i)h(l_j)}$ in the MFJ method, and with probability $\dfrac{\hat{K}_{l_i l_j}}{\max\limits_{1 \le i,j \le L} \hat{K}_{l_i l_j}}$ in MMF method.

If the interaction happens, then:

(a) with the probability $w(l_i)/w(l_i + l_j)$ the cluster l_i is removed,

(b) with the probability $w(l_j)/w(l_i + l_j)$ the cluster l_j is removed,

(c) a new cluster $l_i + l_j$ appeared.

Let us now consider the cost of these algorithms for some fixed N. The difference of the MFJ and MMF algorithms comes mainly from the different choice of the time step (step 1) and different ways of generating the reacting particles (step 2). Let us compare the first step.

Step 1. Obviously, the algorithm MFJ has a larger mean time step if

$$\left(\sum_{i=1}^{L} h(l_i)\right)^2 < L(L-1) \max_{1\le i,j\le L} \hat{K}_{l_i,l_j}.$$

The difference between the left- and right-hand sides is large for the fast growing coagulation coefficients K_{ij}; the inequality can be approximated by $h(i)h(j) \ge K_{ij}$ (for instance, $K_{ij} = ij$). However, this difference can be small for slowly growing coefficients. Not that for bounded coefficients $K_{ij} \le$ const the left- and right-hand sides coincide.

Step 2. Sampling of reacting particles in MMF method is very simple. Namely, the number i, uniformly distributed in $\{1, \dots, L\}$ can be found by the following code (C++, α is uniformly distributed in [0; 1]):

```
long fun;

fun=αL;
if( fun==L ){ i=1; }
else{ i=fun+1; }
```

So, to generate the reacting particle we need two arithmetic operations and one "if" jump. Whilst the generation of the reacting particle in the first algorithm needs in average about of L operations.

Thus we can assume that the first algorithm is preferable in the cases of fast growing coagulation kernels, which can be close approximated with $h(i)h(j) \ge K_{ij}$.

4 Comparative Analysis of the Efficiency

We consider the following algorithms:

- method of majorant frequencies (MMF). As mentioned in Introduction, this method is a modification of the direct simulation algorithm. Its description can be found in [16]
- method of majorant frequencies with doubling of the number of particles, when it becomes two times smaller
- mass-flow algorithm (MFA) (see [4] for the detailed description)

- weighted algorithm (WA), described in the previous section, for the weight function $w(l) = l^q$, with $q = 0.5$, $q = 1$, $q = 1.25$.

Remark 4.1. The weight equation with $w(l) = l$ coincides with the mass-flow equation. However, as we will see below, different algorithms (WA and MFA) for solving this equation provide for fixed N different biases and have different computational costs.

For our investigation we take the pure coagulation equation with monodisperse initial conditions

$$u_1(0) = 1, \quad u_l(0) = 0, \ l > 1,$$

and the following coagulation coefficients:

$$\begin{aligned} &\textbf{(1)} \qquad K_{ij} = 1; \\ &\textbf{(2)} \quad K_{ij} = 0.5(i + j). \end{aligned}$$

These coefficients present examples of slow (case 1) and fast (case 2) coagulation regimes. Analytical solutions to the relevant coagulation equation for these cases read (see, e.g., [1]):

$$\begin{aligned} &\textbf{(1)}: \quad u_l(t) = (1 + 0.5t)^{-2} \left(\tfrac{0.5t}{1+0.5t}\right)^{l-1} \ , \\ &\textbf{(2)}: \qquad u_l(t) = e^{-0.5t} B(1 - e^{-0.5t}, l), \qquad B(x, l) = \frac{(lx)^{l-1} e^{-lx}}{l!}. \end{aligned}$$

Remark 4.2. As we have already mentioned, MMF can be considered as a particular case of a more general "method of fictitious jumps" (MFJ), developed in [5]. However, for the constant coagulation coefficients MFJ coincides with MMF. In the case of the additive coagulation coefficients (2) the calculations show that MMF is the most effective version of MFJ.

Remark 4.3. For additive coefficients only WA with $w(l) = l^{0.5}$ will be considered. Computational cost of WA with $w(l) = l$ and $w(l) = l^{1.25}$ is too high to be included in considerations.

We take the time interval $[0; T]$ with $T = 20$ for the constant coagulation coefficients, and $T = 5$ for the additive coefficients. This choice ensures that the particle concentration at T decreases to about 90%.

Let us introduce the following notations.

N, the number of the monomers in the system at zero time
M, the number of samples, i.e., the independent realizations of the algorithm
τ, the computational cost per one sample.

For $h(u)$, an arbitrary function of the solution to the coagulation equation (3), the following characteristics are defined:

$$\mathbb{E}[h(U)] = \frac{1}{M} \sum_{i=1}^{M} h(U^{(i)}), \quad \text{the mean value taken over } M \text{ samples}$$

$$\mathrm{Var}[h(U)] = \frac{1}{M} \sum_{i=1}^{M} \left(h(U^{(i)})\right)^2 - (\mathbb{E}[h(U)])^2, \quad \text{the mean variance taken over } M \text{ samples}$$

$\mathrm{Var}_{rel}[h(U)] = \dfrac{1}{h^2(u)}\mathrm{Var}[h(U)]$, the relative mean variance realizations;

$\mathrm{Er}[h(U)] = 100\% \times \dfrac{|u_l(t) - \mathbb{E}[h(U)]|}{h(u)}$, the relative error of the algorithm

$\mathrm{St}[h(U)] = 100\% \times 3\sqrt{\dfrac{\mathrm{Var}_{rel}[h(U)]}{M}}$, the relative statistical error of the algorithm

$m_2 = \sum\limits_{i\geq 1} i^2 u_i$, the second moment of the solution.

4.1 Bias and Variance of the Algorithms.

Before analysing the efficiencies of the algorithms, we give several conclusions about the behaviour of their bias and variance.

Conclusion 4.1 *The bias and the variance of $h(u)$ are inverse proportional to N if $h(U)$ is linear, for all algorithms under consideration.*

This conclusion is based on a large series of calculations, made for different functions $h(u)$, e.g., for m_2 and u_l, for different values of l, and for the distribution $c_l = \sum\limits_{i\geq l} u_i$ for different values of l as well. For illustration, we present here the error of the algorithms, when calculating u_2 for different N, for constant (Tables 4.3–4.4) and additive (Tables 4.5–4.6) coagulation coefficients. One can see that the 5-times diminishing of N leads to a 5-times increase of the error. Note that the statistical error is no larger than 10% of the total error of the algorithms, so we can treat $\mathrm{Er}[h(U)]$ as the bias of the estimator.

Another illustration of the Conclusion 4.1 is presented in Tables 4.9–4.12, where the error of the algorithms when calculating the second moment m_2 is shown for constant (Tables 4.9–4.10) and additive (Tables 4.11–4.12) coagulation coefficients. Again, a 2-times enlargement of the value of N leads to a 2-times decrease of the bias and variance of the algorithms.

Let us now compare the error and the variance of the algorithms. According to the Conclusion 4.1, the ratio of the errors and the variance of the different algorithms (which we are interested in) will be the same for all values of N. So, we will consider in detail one particular choice of N. Namely, we take $N = 128$ for constant, and $N = 1024$ for additive coefficients. The number of samples M for different algorithms are chosen so that the largest statistical errors are approximately all the same for all algorithms (which is indicated in the figures).

In Figs. 4.1-4.2 we plot the relative error of algorithms when calculating $u_i(t)$ for different time instances. Namely, the results are given at times, when the total particle concentration becomes 5 times smaller ($t = 8$ for constant and $t = 3$ for additive coefficients) and 10 times smaller ($t = 20$ and $t = 5$, respectively) compared to the initial number of particles. The size spectra presented in these figures contain 95% of the total mass.

One can see that the bias of the doubling procedure, MFA and WA (for fixed N) is smaller than that of MMF. This is more pronounced for fast growing additive coefficients (nearly 5 times in case (1) and about 100 times in case (2)), while the doubling procedure provides approximately the same decrease of the bias in both cases. Note, that the pictures are nearly the same for both time instances. Hence we can make the following conclusion.

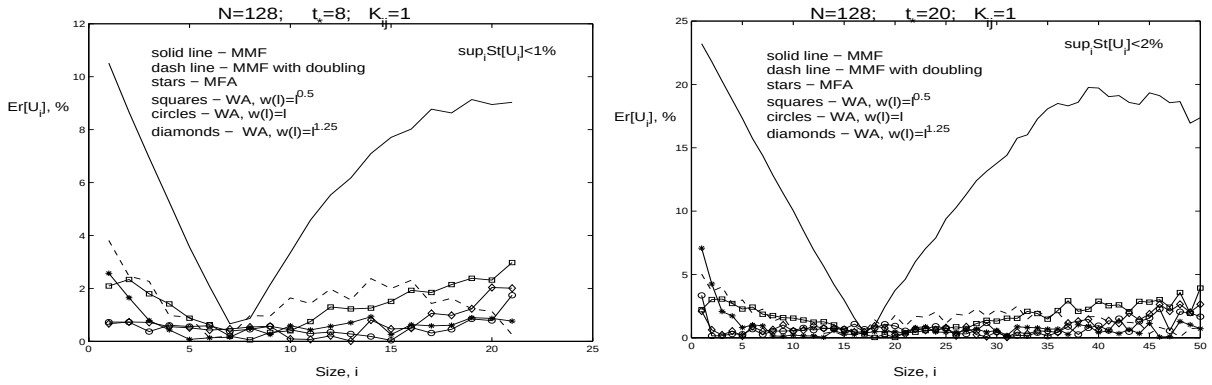


Figure 4.1. The relative error of algorithms: calculations of size distribution $u_i(t_*)$ for $N = 128$ at different times, for constant coagulation coefficient (plotted sizes contain 95% of the total mass).

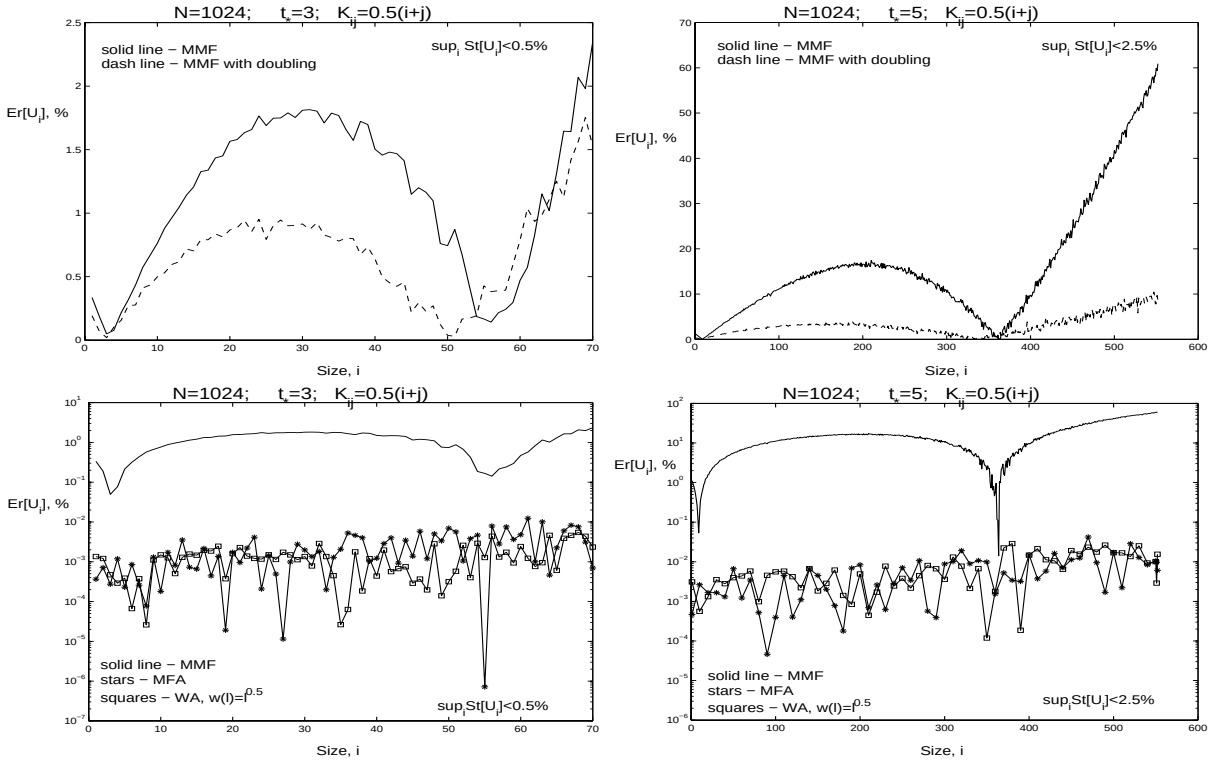


Figure 4.2. The relative error of the algorithms: calculations of size distribution $u_i(t_*)$ for $N = 1024$ at different times, for additive coagulation coefficient (plotted sizes contain 95% of the total mass).

Conclusion 4.2 *The doubling procedure, MFA and WA, all have (N fixed in all algorithms) a bias which is considerably smaller than that of MMF. For MFA and WA this advantage is more pronounced for fast growing coagulation coefficients.*

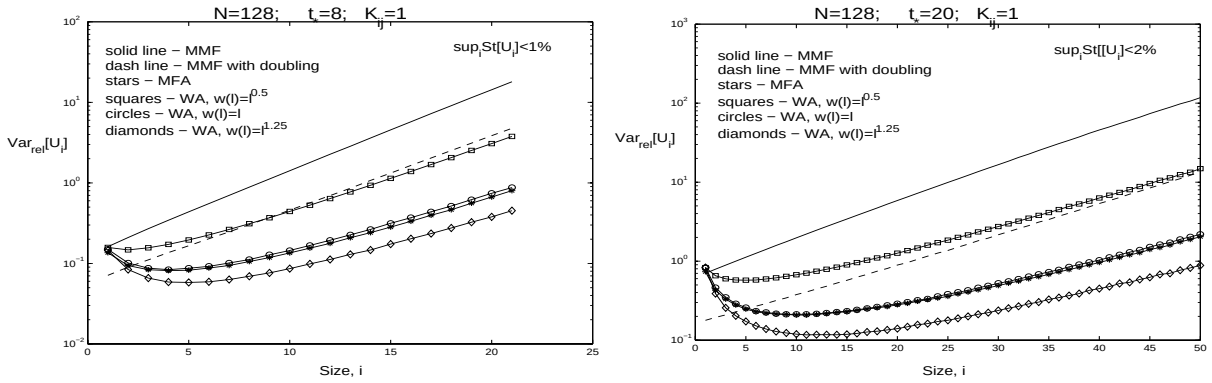


Figure 4.3. The relative variance of the algorithms: calculations of size distribution $u_i(t_*)$ for $N = 128$, at different times, for constant coagulation coefficients (plotted sizes contain 95% of the total mass).

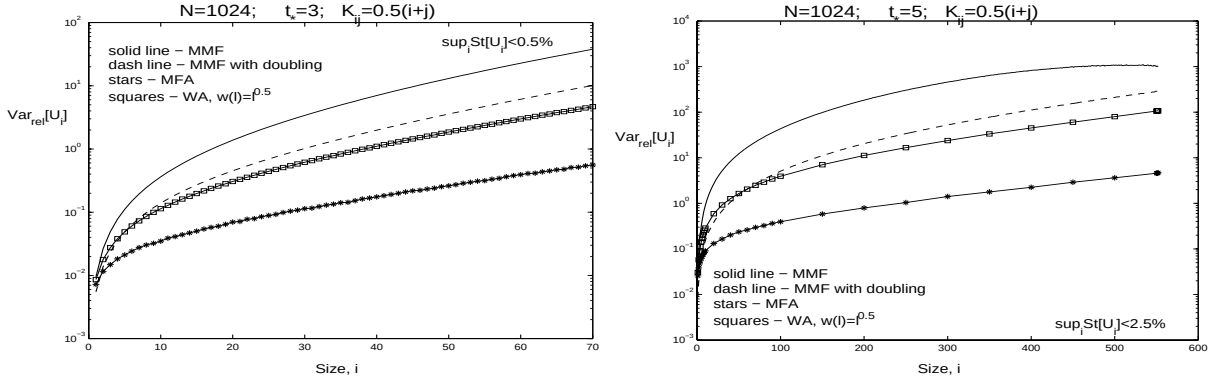


Figure 4.4. The relative variance of the algorithms: calculations of size distribution $u_i(t_*)$ for $N = 1024$ at different times, for additive coagulation coefficients (plotted sizes contain 95% of the whole mass).

Let us now compare the variance of the algorithms for fixed values of N. In Fig. 4.3-4.4 we plot the function $\mathrm{Var}_{rel}[U_i(t_*)]$ at $t_* = 8$ and $t_* = 20$ for constant and at $t_* = 3$ and $t_* = 5$ for additive coefficients. The size spectra, presented in the figures, contain 95% of the total mass.

One can see that in the case of constant coagulation coefficients (Fig. 4.3) the doubling procedure diminishes the variance of MMF approximately 4 times for $t_* = 8$, and 9 times for $t_* = 20$, uniformly with respect to i. The variance of MFA and WA for the monomer concentrations coincides with that of MMF. With the growth of size i, the variance of MFA and WA decreases (up to $i = 5$ for $t_* = 8$, and $i = 12$ for $t_* = 20$). Its further increase is slower than that of MMF variance. Thus for large sizes the MMF has a variance which is considerably larger than the variance of MFA, WA and MMF with doubling. Namely, it is 5 times (when $t_* = 8$) and 8 times (when $t_* = 20$) larger than that of WA with $w(l) = l^{0.5}$; 20 times (when $t_* = 8$) and 60 times (when $t_* = 20$) larger than that of MFA and WA

with $w(l) = l$; 40 times (when $t_* = 8$) and 130 times (when $t_* = 20$) larger than that of WA with $w(l) = l^{1.25}$. However, for small sizes the difference is much smaller. Note that the variance of WA decreases with the growth of q in $w(l)$. For $q = 1$, it coincides with the variance of the MFA.

Let us now turn to the additive coagulation coefficients (Fig. 4.4). One can see that MMF with doubling procedure, MFA and WA again provide a smaller variance compared to that of MMF.

If we consider the ratio of the variances of different algorithms to that of MMF, it turns out that it is again small for small sizes and increases with the growth of i, achieving 4 times (when $t_* = 3$ and $t_* = 5$) for doubling procedure; 68 times (when $t_* = 3$) and 225 times (when $t_* = 5$) for MFA; 8 times (when $t_* = 3$) and 10 times (when $t_* = 5$) for WA with $w(l) = l^{0.5}$. Note that for MFA this effect becomes more pronounced with the growth of time.

Summarizing these results we can make the following conclusion.

Conclusion 4.3. *The doubling procedure, MFA and WA have (the value of N fixed) a lower variance compared to that of MMF. For MFA and WA the difference increases with the growth of t and is more pronounced for the concentration of large particles.*

4.2 Computational Cost

Let us now analyze the relative computational cost of the algorithms for the fixed value of N.

Analysing the description of the algorithms (see [16] for MMF, [4] for MFA and the previous section for WA), we can make the following conclusion concerning the growth of their computational cost with the growth of N.

Conclusion 4.4. *For bounded coagulation coefficients the computational cost of MMF and WA is linear with respect to N, while the computational cost of MFA grows faster than linear. It means that the relative efficiency of MFA compared to MMF and WA, decreases with the growth of the accuracy of the algorithm.*

The nonlinear growth of the computational cost of MFA is caused by the complicated procedure of the choice of the reacting particles. Namely, it is necessary to sample from the distribution of the form

$$\mathrm{P}(i = A_i) = \frac{q_i}{\sum\limits_{j=1}^{N} q_j}, \quad i = 1, \ldots, N,$$

whose computational cost increases with the growth of N.

We illustrate this in Table 4.1. One can see that for MMF and WA, the doubling of N leads to the doubling of the computational cost per one sample. For MFA, this leads to a 3-4 times enlargement of the cost.

Table 4.1. The algorithm cost per one sample (sec.), for $K_{ij} = 1$, $T = 20$.

N	MMF	MMF with doubling	MFA	$w(l) = l^{0.5}$	$w(l) = l$	$w(l) = l^{1.25}$
64	3.67e-5	6.87e-5	2.88e-4	1.81e-4	1.07e-3	3.53e-3
128	7.08e-5	1.35e-4	7.66e-4	3.56e-4	2.09e-3	6.91e-3
256	1.39e-4	2.67e-4	2.29e-3	7.08e-4	4.16e-3	1.38e-3
512	2.75e-4	5.29e-4	7.73e-3	1.41e-3	8.30e-3	2.75e-2
1024	5.44e-4	1.06e-3	2.74e-2	2.81e-3	1.65e-2	5.44e-2
2048	1.09e-3	2.11e-3	1.04e-1	5.64e-3	3.12e-2	1.10e-1
4096	2.18e-3	4.22e-3	3.11e-1	1.12e-2	6.63e-2	2.19e-1
8192	4.36e-3	8.58e-3	1.36	2.25e-2	1.33e-1	4.40e-1

In Table 4.2 we show the computational cost of the algorithms for different values of N in the case of additive coagulation coefficients. It is seen that in this case the cost increases faster than linearly. Namely, the doubling of the value of N leads to a 2.2-2.7 times (MMF and WA, respectively) and 2.2-4 times (MFA) increase of sample's cost.

Table 4.2. The algorithm cost per one sample (sec.), for $K_{ij} = 0.5(i + j)$, $T = 5$.

N	MMF	MMF with doubling	MFA	WA, $w(l) = l^{0.5}$
64	1.72e-4	5.79e-4	7.71e-3	–
128	4.39e-4	1.55e-3	1.72e-2	3.35e-3
256	1.10e-3	4.07e-3	4.12e-2	8.08e-3
512	2.72e-3	1.06e-2	1.11e-2	1.96e-3
1024	6.54e-3	2.59e-2	3.41e-1	4.62e-2
2048	1.56e-2	6.02e-2	1.13	1.10e-1
4096	3.63e-2	1.48e-1	4.09	2.43e-1
8192	8.32e-2	3.40e-1	15.6	5.49

Analysing the comparative computational cost of the algorithms, presented in Tables 4.1-4.2, we can make the following conclusion.

Conclusion 4.4. *The computational cost of MFA and WA for fixed N is considerably larger than that of MMF.*

Let us now discuss the main reasons why the relative cost of MFA and WA is larger compared to MMF, for fixed value of N.

1. MFA: complicated sampling of reacted particles (see the discussion above).
2. Doubling procedure, MFA, and WA: the number of particles in the model system during the whole time behaves essentially different from that of MMF. Indeed, the number of particles in the system in MMF behaves approximately as $\frac{1}{1+0.5t}$ for constant and as $\exp(-0.5t)$ for additive coagulation coefficients, so that at $t = T$ the number of particles is 10 times less. Unlike it, the doubling procedure keeps the number of particles no less than $0.5N$, the MFA keeps N particles for the whole time period, while WA with $q > 1$ even enlarges the number of particles in average.
3. WA: the mean time step is decreased by the factor $\frac{1}{W_{\max}} < 1$ (see (14)).

Summarizing we can conclude, that the enlargement of the relative number of large particles in a model system decreases considerably the bias and variance of the algorithm for fixed N, especially for fast growing coagulation coefficients. However the high computational cost of such algorithms makes their relative efficiency within the framework of the definition (2) not obvious. These algorithms could not be effective when calculating the concentration of small particles, for which their variance does not differ much from that of MMF. They can be efficient for calculating the concentration of large particles, especially for fast growing coagulation coefficients.

To compare the efficiencies of the algorithms, we will present calculations of some functionals of the solution $u(t)$ with some fixed accuracy. Calculations are made for the concentration of small particles, the concentration of large particles and the second moment m_2 of the solution.

Note that in all calculations we will always obtain the same statistical error $||\mathrm{St}[h(U)]||$ for all algorithms. In this particular case we can write for the ratio of efficiencies (2) of two relevant algorithms 1 and 2 that

$$\frac{\mathrm{Eff}_{Alg1}}{\mathrm{Eff}_{Alg2}} = \frac{\tau_{Alg2} M_{Alg2}}{\tau_{Alg1} M_{Alg1}} \ .$$

Thus when comparing efficiencies of algorithms we can use simply the computational cost τM instead of the value $||\mathrm{Var}[U^N(t)]||\tau$.

4.3 Examples

4.3.1 Concentration of Small Particles

In Tables 4.3-4.6 the results of $u_2(t)$-calculation are presented. We compare 6 different algorithms, the relative error is about 1% for constant (Tables 4.3 and 4.4) and 5% for additive (Tables 4.5 and 4.6) coagulation coefficients.

Table 4.3. Calculation of u_2 with the accuracy of about 1%, for $K_{ij} = 1$, $T = 20$.

Algorithm	N	M	$\sup_{t\in[0;T]} \mathrm{Er}[U_2]$	$\sup_{t\in[0;T]} \mathrm{St}[U_2]$	τ(sec.)	τM
MMF	3300	400 000	0.911%	0.093%	1.80e-3	720
MMF, doubling	600	400 000	0.912%	0.106%	6.66e-4	266
MFA	500	1 000 000	1.004%	0.101%	7.93e-3	7932
WA, $w(l) = l^{0.5}$	390	2 300 000	0.975%	0.093%	1.09e-3	2513
WA, $w(l) = l$	110	5 000 000	1.043%	0.098%	1.80e-3	9015
WA, $w(l) = l^{1.25}$	100	4 500 000	0.949%	0.099%	5.49e-3	24687

Table 4.4. Calculation of u_2 with the accuracy of about 5%, for $K_{ij} = 1$, $T = 20$.

Algorithm	N	M	$\sup_{t\in[0;T]} \mathrm{Er}[U_2]$	$\sup_{t\in[0;T]} \mathrm{St}[U_2]$	τ(sec.)	τM
MMF	660	80 000	4.348%	0.458%	3.54e-4	28
MMF, doubling	120	80 000	3.859%	0.483%	1.26e-4	10
MFA	100	200 000	4.916%	0.497%	6.05e-4	121
WA, $w(l) = l^{0.5}$	78	460 000	5.307%	0.454%	2.19e-4	100
WA, $w(l) = l$	22	1 000 000	5.115%	0.490%	3.84e-4	365
WA, $w(l) = l^{1.25}$	20	900 000	4.666%	0.499%	1.18e-3	1065

Table 4.5. Calculation of u_2 with the accuracy of about 1%, for $K_{ij} = 0.5(i + j)$, $T = 5$.

Algorithm	N	M	$\sup_{t\in[0;T]} \mathrm{Er}[U_2]$	$\sup_{t\in[0;T]} \mathrm{St}[U_2]$	τ(sec.)	τM
MMF	1024	850 000	0.932%	0.092%	6.55e-3	5566
MMF, doubling	250	900 000	0.948%	0.092%	3.94e-3	3546
MFA	70	6 000 000	0.992%	0.091%	8.56e-3	51354
WA, $w(l) = l^{0.5}$	315	2 000 000	1.015%	0.092%	1.06e-2	21220

Table 4.6. Calculation of u_2 with the accuracy of about 5% for $K_{ij} = 0.5(i + j)$, $T = 5$.

Algorithm	N	M	$\sup_{t\in[0;T]} \mathrm{Er}[U_2]$	$\sup_{t\in[0;T]} \mathrm{St}[U_2]$	τ(sec.)	τM
MMF	206	170 000	5.575%	0.452%	8.28e-4	141
MMF, doubling	50	180 000	4.879%	0.457%	4.06e-4	73
MFA	14	1 200 000	5.285%	0.443%	1.56e-3	1871

Analysing the last column of the Tables, one can conclude that for this particular problem, MMF with the doubling procedure is the most efficient algorithm. The doubling procedure increases the efficiency of MMF more than 3 times for constant, and more than 1.5 times for additive coefficients.

MFA is considerably less efficient than MMF, for both constant and additive coefficients, and this effect is even more pronounced for higher accuracies, e.g., for constant coefficients we got the following results: at 5%-accuracy level MMF is 4 times faster than MFA, while at 1%-accuracy level it is 11 times faster.

The WA is also less efficient than MMF in this case. For constant coefficients its efficiency decreases with the growth of the power of $w(l)$.

4.3.2 Concentration of Large Particles

Here we will deal with the largest particles of the size spectrum from Figs. 4.1-4.4, for fixed times. Namely, for constant coagulation coefficients we take the concentration of $\{21\}$-clusters at $t = 8$, and $\{50\}$-clusters at $t = 20$ (see Figs. 4.1, 4.3); for additive coagulation coefficients we consider the concentration of $\{70\}$-clusters at $t = 3$ (see Figs. 4.2, 4.4).

Note that the particle subsystem whose size spectrum is presented in these figures contains 95% of the total mass, the largest particles of this size distribution can be considered as large particles.

In Tables 4.7-4.9 we present the results of calculations of $u_{21}(8)$ with the accuracy 3% and $u_{50}(20)$ with the accuracy 5% for constant coefficients, and $u_{70}(3)$ with the accuracy 3% for additive coefficients.

Table 4.7. Calculation of $u_{21}(8)$ with the accuracy of about 3%, for $K_{ij} = 1$.

Algorithm	N	M	$\sup_{t\in[0;T]} \mathrm{Er}[U_{21}(8)]$	$\sup_{t\in[0;T]} \mathrm{St}[U_{21}(8)]$	τ(sec.)	τM
MMF	340	6 500 000	3.124%	0.298%	1.84e-4	1199
MM, doubling	24	25 000 000	3.003%	0.294%	2.83e-5	707
MFA	35	2 800 000	3.146%	0.310%	1.35e-5	379
WA, $w(l) = l^{0.5}$	120	3 800 000	3.001%	0.309%	3.36e-4	1278
WA, $w(l) = l$	80	1 400 000	2.872%	0.302%	1.32e-3	1850
WA, $w(l) = l^{1.25}$	70	800 000	2.961%	0.305%	3.84e-3	3076

Table 4.8. Calculation of $u_{50}(20)$ with the accuracy of about 5%, for $K_{ij} = 1$.

Algorithm	N	M	$\sup_{t\in[0;T]} \mathrm{Er}[U_{50}(20)]$	$\sup_{t\in[0;T]} \mathrm{St}[U_{50}(20)]$	τ(sec.)	τM
MMF	400	2 700 000	5.053%	1.058%	2.17e-4	586
MMF, doubling	20	7 500 000	4.618%	0.989%	2.36e-5	177
MFA	24	1 000 000	4.897%	1.006%	8.87e-5	89
WA, $w(l) = l^{0.5}$	90	1 950 000	5.024%	0.991%	2.74e-4	534
WA, $w(l) = l$	68	372 000	5.128%	1.009%	1.14e-3	423
WA, $w(l) = l^{1.25}$	78	135 000	4.690%	0.998%	4.45e-3	600

Table 4.9. Calculation of $u_{70}(3)$ with the accuracy of about 3%, for $K_{ij} = 0.5(i + j)$.

Algorithm	N	M	$\sup_{t\in[0;T]} \mathrm{Er}[U_{70}(3)]$	$\sup_{t\in[0;T]} \mathrm{St}[U_{70}(3)]$	τ(sec.)	τM
MMF	802	48 500 000	3.091%	0.298%	4.83e-3	234027
MMF, doubling	600	17 500 000	3.089%	0.298%	1.18e-2	205677
MFA	48	1 200 000	2.951%	0.296%	5.65e-3	6781
WA, $w(l) = l^{0.5}$	136	34 500 000	2.952%	0.299%	3.90e-3	134670

One can see that for this problem, MFA is the most efficient among the considered algorithms. Its relative efficiency is more pronounced in the case of the fast growing additive coefficients and the efficiency achieves a 36-times advantage compared to MMF.

The doubling procedure improves MMF for constant coefficients, and the advantage increases with time: 1.7 times when calculating $u_{21}(8)$ and 3.3 times when calculating $u_{50}(20)$. However for additive coefficients the efficiency of MMF practically does not change when using the doubling procedure.

The relative efficiency of WA, compared to MMF, also increases with the growth of t in the case of the constant coefficients. This effect is more pronounced for larger powers of $w(l)$. So, WA with $w(l) = l^{1.25}$ is practically as efficient as MMF when calculating $u_{50}(20)$, while it is approximately 2.5 times less efficient for $u_{21}(8)$.

4.3.3 Second Moment of Solution

In Tables 4.10-4.13 we present the results of calculation of the second moment m_2 with the accuracy of about 1% and 2% for constant coefficients (Tables 4.10 and 4.11), and of about 2.5% and 5% for additive coefficients (Tables 4.10 and 4.11).

Table 4.10. Calculation of m_2 with the accuracy of about 1%, for $K_{ij} = 1$, $T = 20$.

Algorithm	N	M	$\sup_{t\in[0;T]} \mathrm{Er}[m_2]$	$\sup_{t\in[0;T]} \mathrm{St}[m_2]$	τ(sec.)	τM
MMF	700	180 000	0.995%	0.099%	4.15e-4	75
MMF, doubling	160	150 000	0.967%	0.097%	1.78e-4	27
MFA	64	80 000	1.040%	0.095%	3.46e-4	28
WA, $w(l) = l^{0.5}$	190	400 000	0.959%	0.099%	4.98e-4	199
WA, $w(l) = l$	128	400 000	1.065%	0.100%	2.10e-3	840
WA, $w(l) = l^{1.25}$	110	410 000	1.136%	0.101%	5.97e-3	2447

Table 4.11. Calculation of m_2 with the accuracy of about 2% for $K_{ij} = 1$, $T = 20$.

Algorithm	N	M	$\sup_{t\in[0;T]} \mathrm{Er}[m_2]$	$\sup_{t\in[0;T]} \mathrm{St}[m_2]$	τ(sec.)	τM
MMF	350	90 000	1.894%	0.197%	1.90e-4	17
MMF, doubling	80	75 000	1.987%	0.190%	8.52e-5	6
MFA	32	40 000	2.231%	0.193%	1.22e-4	5
WA, $w(l) = l^{0.5}$	95	200 000	1.885%	0.201%	2.65e-4	53
WA, $w(l) = l$	64	200 000	2.036%	0.204%	1.07e-3	214
WA, $w(l) = l^{1.25}$	55	205 000	2.197%	0.205%	3.06e-3	627

Table 4.12. Calculation of m_2 with the accuracy of about 2.5% for $K_{ij} = 0.5(i + j)$, $T = 5$.

Algorithm	N	M	$\sup_{t\in[0;T]} \mathrm{Er}[m_2]$	$\sup_{t\in[0;T]} \mathrm{St}[m_2]$	τ(sec.)	τM
MMF	14000	40 000	2.607%	0.320%	1.59e-1	6375
MMF, doubling	3000	60 000	2.283%	0.263%	9.63e-2	5778
MFA	24	540 000	2.547%	0.247%	2.70e-3	1460
WA, $w(l) = l^{0.5}$	300	2 600 000	1.827%	0.145%	9.98e-3	25958

Table 4.13. Calculation of m_2 with the accuracy of about 5% for $K_{ij} = 0.5(i+j)$, $T = 5$.

Algorithm	N	M	$\sup_{t\in[0;T]} \mathrm{Er}[m_2]$	$\sup_{t\in[0;T]} \mathrm{St}[m_2]$	τ(sec.)	τM
MMF	7000	20 000	4.995%	0.571%	7.01e-2	1403
MMF, doubling	1500	30 000	4.743%	0.493%	4.08e-2	1223
MFA	12	270 000	5.038%	0.514%	1.35e-3	364
WA, $w(l) = l^{0.5}$	150	1 300 000	4.523%	0.417%	4.03e-3	5245

One can see that for constant coefficients, the efficiencies of MMF with the doubling procedure and MFA are practically the same: they are approximately 3 times higher than that of MMF. In contrast, for additive coefficients, the doubling procedure practically does not change the efficiency of MMF, while the efficiency of MFA is approximately 4 times higher compared to MMF.

The efficiency of WA is in this case lower compared to the other considered algorithms.

4.4 Discussion and open problems

Summarizing the discussion above we give the following remarks.

1. *For bounded coagulation coefficients it is better to use MMF with the doubling procedure. But if it is necessary to obtain the concentration of large particles, MFA is preferable.*

2. *For fast growing coagulation coefficients ($K_{ij} \geq C(i+j)$) MFA is preferable, but not in the case when only the concentration of small particles is important: then MMF with the doubling procedure is more efficient.*

This conclusion leaves however some important problems open. For instance, it is clear that the relative efficiencies of the algorithms depend generally on the structure of the coagulation coefficients. We can consider the bounded coefficients as a characteristic example of the "slowly growing" coefficients, and $K_{ij} \geq C(i+j)$ as characterizing the "fast growing" coefficients. But we cannot predict for sure that the situation for moderate changing coefficients will be similar.

A question which arises in the context of the small- and large particle parts of the size spectrum: which particles can be treated as "large" and "small" in particular problems? Obviously it depends on many physical features of the coagulation-fragmentation system, and on the definition of the monomer as well.

Finally, very important open problem is the decrease of the computational cost of implementation of MFA and WA. As we have shown, these methods have sufficiently small bias and variance for fixed initial number of particles in the model system, but their high computational costs makes their competition with other methods problematic.

5 Coagulation and Evaporation in Free Molecule Regime

Consider a coagulation-evaporation process in a free molecule regime, which is governed by the equation (see [6])

$$\frac{\partial u_1(t)}{\partial t} = -u_1 \sum_{i\geq 1} K_{i1} u_i + \sum_{i\geq 3} E_i u_i + 2E_2 u_2, \tag{15}$$

$$\frac{\partial u_l(t)}{\partial t} = \frac{1}{2} \sum_{i+j=l} K_{ij} u_i u_j - u_l \sum_{i\geq 1} K_{li} u_i - E_l u_l + E_{l+1} u_{l+1}, \quad l \geq 2$$

with

$$K_{ij} = \left(\frac{3}{4\pi}\right)^{\frac{1}{6}} \left(\frac{6kT}{\rho_p}\right)^{\frac{1}{2}} \left(\frac{1}{i} + \frac{1}{j}\right)^{\frac{1}{2}} \left(i^{\frac{1}{3}} + j^{\frac{1}{3}}\right)^2;$$

$$E_j = EK_{1j} \exp\left(A\left(j^{\frac{2}{3}} - (j-1)^{\frac{2}{3}}\right)\right).$$

Here k is Boltzmann's constant, T is the absolute temperature, ρ_p is the particle density.

In what follows we will suppose monodisperse initial conditions:

$$u_1(0) = u^{(0)}; \quad u_l(0) = 0,\ l > 1.$$

The question we are interested in is how the values of the evaporation parameters influence the behaviour of the process. To be more specific, we study the total concentration of all particles whose sizes are larger than a given size j_*: $c_{j_*}(t) = \sum_{i\geq j_*} u_i(t)$.

We will first write the equation (15) in more convenient form. Note that the solution to (15) depends on large number of parameters. Actually, u_i is a function of t, $u^{(0)}$, k, T, ρ_p, A, and E. Our aim now is to reduce the number of parameters.

Let us introduce a characteristic time scale of the coagulation process

$$T_C = \frac{1}{u^{(0)} K_{11}},$$

which can be interpreted as the mean collision time of the particles at the initial time. Then let

$$T_E = \frac{1}{E_2}$$

be the characteristic time scale of the evaporation, and

$$\beta = \frac{T_C}{T_E} \tag{16}$$

the ratio of these time scales. Now, using the dimensionless time

$$\tau = \frac{t}{T_C} \tag{17}$$

we rewrite the equation (15) as follows:

$$\frac{\partial u'_1(\tau)}{\partial \tau} = -u'_1 \sum_{i\geq 1} k_{i1} u'_i + \sum_{i\geq 3} e_i u'_i + 2e_2 u'_2, \tag{18}$$
$$\frac{\partial u'_l(\tau)}{\partial \tau} = \frac{1}{2} \sum_{i+j=l} k_{ij} u'_i u'_j - u'_l \sum_{i\geq 1} k_{li} u'_i - e_l u'_l + e_{l+1} u'_{l+1}, \quad l \geq 2$$

where

$$u'_l = \frac{u_l}{u^{(0)}},$$

$$k_{ij} = \frac{K_{ij}}{K_{11}} = \frac{1}{4\sqrt{2}} \left(i^{\frac{1}{3}} + j^{\frac{1}{3}}\right)^2 \left(\frac{1}{i} + \frac{1}{j}\right)^{\frac{1}{2}},$$

$$e_j = \beta \exp\left(A\left((j^{\frac{2}{3}} - (j-1)^{\frac{2}{3}}) - (2^{\frac{2}{3}} - 1)\right)\right).$$

It implies the following assertion:

Statement 5.1.
The function $u(t;\ u^{(0)}, k, T, \rho_p, A, E)$ *can be represented as follows:*

$$u(t;\ u^{(0)}, k, T, \rho_p, A, E) = u^{(0)} u'(\tau;\ \beta, A),$$

where β *and* τ *are defined by (16) and (17),* u' *being the solution to (18).*

Let us now study how the change of the evaporation parameters A and β influence the behaviour of the function $c'_{j_*}(\tau)$ for different j_*.

It is worth to be mentioned that the following processes change the value of the total concentration of particles whose size is larger than j_*.

1. Collisions between two particles whose sizes are less than j_* can result in particles of size, larger than j_*; this gives a contribution to the increase of value of $c'_{j_*}(\tau)$.

2. Coagulation of two particles whose sizes are larger than j_* decrease the value of $c'_{j_*}(\tau)$.

3. Evaporation of monomers: it decreases the value of $c'_{j_*}(\tau)$ and can be considered as an effective time-dependent monomer generator.

Simple analysis shows that two different regimes should exist for $c'_{j_*}(\tau)$. First, in the initial time interval the total concentration of particles larger than j_* increases. It is caused by the prevailing role of the collision processes among smaller particles, since initially, only monomers existed. After achieving a maximal value, the function $c'_{j_*}(\tau)$ decreases due to the increasing role of the collisions between particles larger than j_*.

In Fig. 5.1 we plot the function $c'_{j_*}(\tau)$ for different ratios β, keeping the parameter A fixed ($A = 1$). Note that the case $\beta = 0$ corresponds to the pure coagulation, because the equation (18) then obviously corresponds to the equation (15) with $E_j = 0$, $j \geq 2$. In the terms of the time scales this situation can be interpreted as the infinite evaporation time scale T_E. The increase of β corresponds to the relative decrease of the evaporation time scale T_E.

The increase of β damps down the growth of $c'_{j_*}(\tau)$ at the first stage, and the maximum value is getting smaller. This is caused by the evaporation of the monomers: its influence is essential when the intensive growth of particles via monomers takes place.

Let us consider the second stage. In the pure coagulation case, the number of particles smaller than j_* becomes now relatively small. So, the process is determined by the collisions between the larger particles. However, the evaporation gives an additional permanent generator of monomers, which, in turn, causes generation of larger particles. Thus, the value of $c'_{j_*}(\tau)$ becomes larger, compared to the pure coagulation case. The difference increases with the growth of β. Note, that for $\beta \geq 1$ the function $c'_{j_*}(\tau)$ is practically constant in the second stage.

Let us now consider how the total concentration of particles, larger than j_* depends on the evaporation parameter A. In Fig. 5.2 we plot the function $c'_{j_*}(\tau)$ for different values of A, keeping $\beta = 1$. One can see that the increase of the parameter A leads to the same effect, as the decrease of the parameter β (see Fig. 1). Namely, the growth of $c'_{j_*}(\tau)$ in the first stage and its decrease in the second stage become slower, and its maximal value decreases. The reason of this is that the decrease of the parameter A leads to the increase of the evaporation coefficients e_j, since $(j^{2/3} - (j-1)^{2/3}) - (2^{2/3} - 1) < 0$ for each $j > 2$. This implies the relative growth of the intensity of evaporation compared to coagulation.

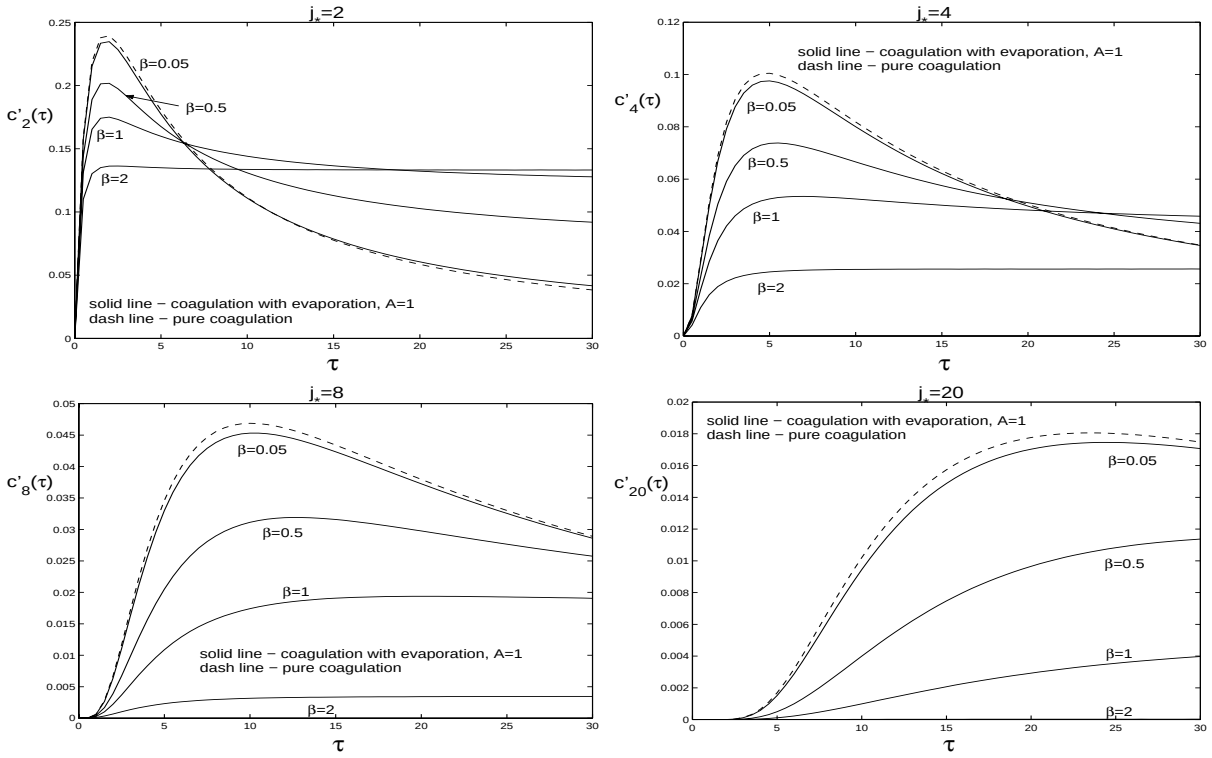


Figure 5.1. The summary concentration of the particles, larger, than j_* via dimensionless time τ for $A = 1$ and different values of β.

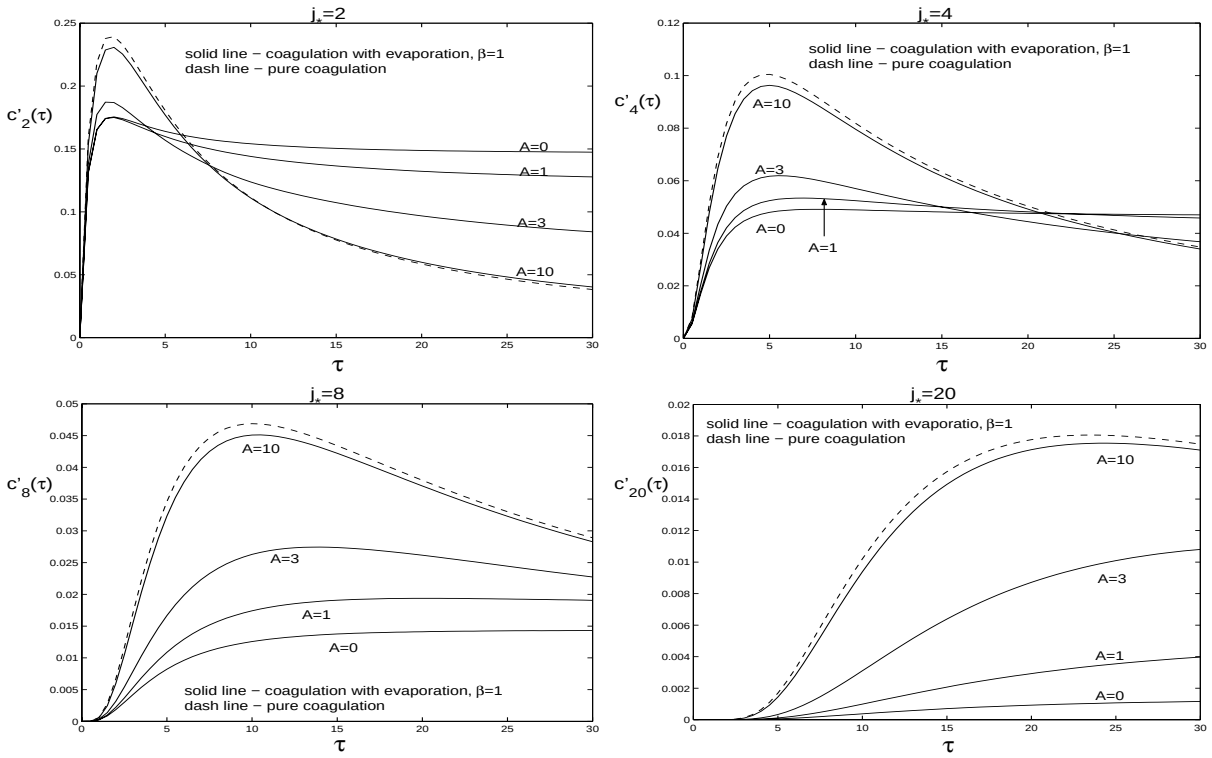


Figure 5.2. The summary concentration of the particles, larger, than j_* via dimensionless time τ for $\beta = 1$ and different values of A.

6 Conclusion

A general variance reduction technique for stochastic particle methods for solving the coagulation-fragmentation Smoluchowski equation is suggested. This method generalizes the mass-flow approach proposed by H. Babovsky, and is aimed at variance reduction for a specific band of the size spectrum. Estimations of the variance and bias of the method are derived. A comparative cost and variance analysis is made for the known stochastic methods. An applied problem of coagulation-evaporation dynamics in free molecule regime is solved.

We have shown, that our approach, as well as the mass-flow algorithm, provides considerably smaller bias and variance of the estimator, compared with the direct simulation algorithm. This advantage is more pronounced for the fast growing coagulation coefficients, when calculating the concentration of large particles. However, the high computational cost of these algorithms makes their competition with other algorithms not obvious. Namely, they proved to be preferable only in the case of fast growing coagulation coefficients, when calculating the concentration of large particles.

We have shown, that a simple doubling of the particles in the model system (after their number becomes 2 times smaller) reduces the bias and the variance of the direct simulation algorithm and increases its efficiency up to 2 times. The direct simulation algorithm with the doubling procedure proved to be the most efficient among the other algorithms for bounded coagulation coefficients and when calculating the concentration of small particles.

References

[1] Aldous D.J. Deterministic and Stochastic Models for Coalescence (Aggregation and Coagulation): a Review of the Mean-Field Theory for Probabilities. *Bernoulli* (1999), **5**, No.1, 3–48.

[2] Babovsky H. On the Monte Carlo Scheme for Smoluchowski Coagulation Equation. *Monte Carlo Methods and Appl.* (1999), **5**, No.1, 1–18.

[3] Ball J.M., Carr J. The Discrete Coagulation-Fragmentation Equations: Existence, Uniqueness and Density Concentration. *J. Stat. Phys.* (1990), **61**, 203–234.

[4] Eibeck A., Wagner W. Stochastic Particle Approximation for Smoluchowski's Coagulation Equation. *Ann. Appl. Prob.* (2001), **11**, No.4, 1137-1165.

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

[6] Friedlander S.K. *Smoke, dust and haze.* J. Wiley & Sons, 1977.

[7] Gillespe D.T. An Exact Method for Numerically Simulating the Stochastic Coalescence Process in a Cloud. *J. Atm. Sci.* (1975), **32**, 1977–1989.

[8] Guias F. A Monte Carlo Approach to the Smoluchowski Equation. *Monte Carlo Methods and Appl.* (1997), **3**, No.4, 313–326.

[9] Ivanov M.S., Rogasinsky S.V. *The Direct Statistical Simulation Method in Rarefied Gas Dynamics.* Nauka, Computing Centre, Novosibirsk, 1988 (in Russian).

[10] Goodson M., Kraft M. An Efficient Stochastic Algorithm For Simulating Nanoparticle Dynamics. *J. Comp. Phys.* (2002), **183**, 210–232.

[11] Kolodko A., Wagner W. Convergence of a Nanbu Type Method for the Smoluchowski Equation. *Monte Carlo Methods and Appl.* (1997), **3**, No.4, 255–273.

[12] Liffman K. A Direct Simulation Monte Carlo Method for Cluster Coagulation. *J. Comp. Phys.* (1992), **100**, 116–127.

[13] Lushnikov A.A. Some New Aspects of Coagulation Theory. *Izv. Acad. Nauk SSSR*, (1978), **14**, No.10, 738–743 (in Russian).

[14] Marcus A.H. Stochastic Coalescence. *Technometrics* (1968), **10**, No.1, 133–148.

[15] Nanbu K. Direct Simulation Scheme derived from Boltzmann Equation. I. Monocomponent Gases. *J. Phys. Soc. of Japan* (1980), **49**, No.5, 2042–2049.

[16] Sabelfeld K.K., Rogasinsky S.V., Kolodko A.A., Levykin A.I. Stochastic Algorithms for Solving Smolouchovsky Coagulation Equation and Applications to Aerosol Growth Simulation. *Monte Carlo Methods and Appl.* (1996), **2**, No.1, 41–87.

[17] Sabelfeld K.K. and Kolodko A.A. Monte Carlo simulation of the coagulation processes governed by Smoluchowski equation with random coefficients. *Monte Carlo Methods and Appl.* (1997), **3**, No.4, 275–311.

[18] Voloshtchuk V.M., Sedunov Ju.S. *The Coagulation Processes in Dispersed Systems.* Gidromet., Leningrad, 1975. (in Russian)

[19] Williams M.M.R. and Loyalka S.K. *Aerosol Science. Theory and Practice.* Pergamon, New York, 1991.