

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

6,900

Open access books available

186,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

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

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

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



Monte Carlo Simulation of Coagulating Aerosols

Jose Ignacio Huertas

Additional information is available at the end of the chapter

<http://dx.doi.org/10.5772/62021>

Coagulation describes the process by which particles that are suspended in a fluid come into contact and coalesce or adhere to one another. Particles interact because of Brownian (thermal) motion, hydrodynamics, and/or gravitational, Van der Waals, Coulomb, or other forces. During this work, only coagulation due to Brownian motion is considered as this is the dominant mechanism for particle interactions in aerosols. Coagulation due to Brownian motion has been extensively studied and expressions for the rate of coagulation are well known. Furthermore, these expressions have been validated experimentally. [64]

During the development of these expressions it is assumed that upon collision, particles stick together and form a third single spherical particle whose volume is equal to the sum of the original two, i.e., the assumptions are:

- Spherical particles. This assumption implies that the rate of sintering or coalescence is instantaneous compared to the rate of collision. This is a good assumption for very small particles, as shown in Appendix B.
- Accommodation or sticking coefficient of unity. Although little is known quantitatively about the accommodation coefficient of aerosol particles, their low kinetic energies make bounce-off unlikely.

The rate of collision (Z_{ij}) between particles of size v_i and v_j is given by $Z_{ij} = \beta_{ij} n_i n_j$, where n_i is the number of particles of size $v_{i-1} < v < v_i$ per unit volume, and β_{ij} is the collision frequency function. Table 1 gives expression for β_{ij} in the free molecular regime (FMR) and in the continuum regime (CR), when coagulation is due to Brownian motion.

For a monodisperse aerosol with N particles per unit volume, the collision frequency simplifies to $Z = BN^2/2$. Then, the characteristic time for coagulation $\tau_{\text{coll}} = N/Z = 2/\beta N$, where τ_{coll} is the average elapsed time between successive collisions for a particle of size v . It can also be interpreted as the time required to halve the particle concentration of an initially monodisperse aerosol by coagulation, or the time to double the volume of the particles. Table 1 shows expressions for τ_{coll} .

	Collision Frequency Function	Characteristic Time for Coagulation
	β_{ij}	$\tau_{\text{coll}} [\text{s}]$
FRM	$\left(\frac{6KT}{\rho_p}\right)^{\frac{1}{2}} \left(\frac{1}{r_i^3} + \frac{1}{r_j^3}\right)^{\frac{1}{2}} (r_i + r_j)^2$	$\frac{1}{4N} \sqrt{\frac{\rho}{3KT}}$
CR	$\left(\frac{2KT}{3\mu}\right) \left(\frac{1}{r_i^3} + \frac{1}{r_j^3}\right)^{\frac{1}{2}} (r_i + r_j)^2$	$\frac{3\mu}{4KT}$

Table 1. Expressions for the collision frequency function in the free molecular regime (FMR) and continuum regime (CR). [64]

The theory of coagulation is essentially a scheme for keeping account of particle collision as a function of particle size. The rate of formation of particles of size k by collision of particles of size i and j is given by $\frac{1}{2} \sum_{i+j=k} Z_{ij}$ where the notation $i+j=k$ indicates that the summation is taken over those collisions for which $v_k = v_i + v_j$. The factor $1/2$ is introduced because each collision is counted twice in the summation. The rate of loss of particles of size k by collision with all other particles is $\sum_j Z_{kj}$. Hence, the net rate of generation of particles of size k is given by the expression:

$$\frac{dn_k}{dt} = \frac{1}{2} \sum_{i+j=k} \beta_{ij} n_i n_j - n_k \sum_{j=1}^{\infty} \beta_{kj} n_j \quad (1)$$

which is the dynamic equation for the discrete size spectrum when only coagulation is important. The solution of Equation (1) depends on the form of β_{ij} , which is determined by the kinetics of particle collision. In this chapter, the Monte Carlo method developed to solve Equation (1) is described. Initially, a general description of the Monte Carlo technique is presented and then the method is applied to solve the coagulation equation. Later, the model is validated by comparing results with analytical and numerical solutions and experimental observations.

1. Monte Carlo simulation

The Monte Carlo technique is a method of statistically sampling events to determine the average behavior of a system. [65] The name Monte Carlo was first applied to a class of mathematical methods by scientists working on the development of nuclear weapons in Los Alamos in the 1940s. The essence of those methods was the invention of games of chance whose behavior and outcome were used to study statistical and deterministic phenomena. Since then, the term Monte Carlo has been employed to describe a wide range of procedures involving the notion of sampling.

Historically, the first example of a computation by a Monte Carlo method was reported by Buffon in 1777, who described an unexpected method to calculate the value of π . [66] With the increasing availability of very high speed general-purpose computers, many problems have become treatable and the Monte Carlo method has increased in popularity. The technique has

been applied in many disciplines to simulate very diverse systems. The greatest successes of the Monte Carlo method have arisen in areas where the problem itself consists of some random process. It has been applied to simulate problems like chemical kinetics, diffusion, radiation, turbulence, and air pollution.

The solutions obtained through a Monte Carlo simulation are statistical in nature and subject to the laws of chance. This aspect is a drawback, but not a serious one. Convenience, ease, directness, and expressiveness of the method are important assets. Other general features of the method are:

- Cyclic nature of the programming
- Necessity of performing a large series of computations of a uniform type
- Use of a comparatively small amount of memory for storage of intermediate results
- Smoothing of errors

A distinction is sometimes made between *direct simulation* and *Monte Carlo simulation*. In this view, the *direct simulation* is a rather direct transcription into computing terms of a natural stochastic process. *Monte Carlo simulation*, by contrast, is the solution by probabilistic methods of nonprobabilistic problems. The distinction is somewhat useful but often impossible to maintain.

The *direct simulation* replaces the mathematical analysis by a computational algorithm in which the approximations due to the requirements of the mathematical analysis can be avoided. The study of dynamics of gases composed of hard spheres is the classical example. With the direct simulation, the trajectories of molecules are followed by numerically solving the classical equation of motion to predict the location of the particles at successive short intervals of time. At each interval, the system is examined to discover whether collisions have occurred. After every collision, a suitable set of new velocities is given to the particles involved, and the process continues. The changes in velocity distribution, for example are followed by recording the velocity of every particle through many cycles. This application is typical of many in the molecular sciences in that the individual processes (in this case collision) may be quite well understood, but the bulk or average properties cannot be readily related to these events.

With the *Monte Carlo simulation*, the physical problem is replaced by an artificial, probabilistic one. The solution of chemical reaction equations [67] can be used as an illustrative example. In this case, a reactive molecule is represented in the computer by the digit 1. Reaction is indicated by replacing 1's by 0's. To start the reaction, the different storage portions representing each type of molecule are filled with 1's according to their initial concentration. At each step, a random number is generated for each type of molecule being considered. In a second-order reaction, for example, if the numbers 75 and 85 are generated, the 75th molecule type A reacts with the 85th molecule type B by replacing the 1's in each location by 0's. If one or both locations already contain 0's, no reaction occurs. Higher-order reactions, competing reactions, and sequential reactions have been simulated through this technique as well.

1.1. Monte Carlo simulation of aerosol processes

The Monte Carlo method has also been used in the study of aerosol dynamics. Kaplan et al. [68] studied the agglomeration of chain-like combustion aerosols due to Brownian motion. They used the so-called *ising or lattice model*, which has been extensively used in polymer science. This model is a self-avoiding random walk on a lattice, where the state of a particular point in the system depends on the states of its nearest neighbors.

The basic algorithm is as follows: At the beginning of each simulation, monodisperse or lognormally distributed clusters are randomly distributed on the corners of a 2D or 3D lattice. During each time step, each cluster moves at random toward any of the 4 (2D) or 6 (3D) possible directions on the lattice. The probability of movement of a given cluster is dependent upon cluster size as predicted by the diffusion theory; smaller clusters have a proportionately higher probability of diffusion compared to large ones. The probabilities of movement are scaled by assuming the probability of movement of a single mer at, for example, 2000 K to be 1. Clusters located on adjacent nodes have a probability of collision that is dependent upon the form of the collision frequency function. At the end of each time step, after each particle has been displaced, and all possible collisions have been checked, the particle size distribution is updated, and the simulation continues until the required time duration is reached. This model accounts only for coagulation, assumes cluster–cluster collisions to be irreversible, and neglects sintering.

Akhtar et al. [69] advanced this work and performed a Monte Carlo simulation of the gas-phase coagulation and sintering of nanosized particles. Particle coagulation and growth were simulated on a 500 x 500 square lattice with periodic boundary conditions to reduce edge effects. During the sintering routine, a particle on any of the perimeter sites of the aggregate is picked at random and allowed to move to any neighboring unoccupied perimeter site on the same aggregate. The probability of acceptance of such movement is determined by the total potential of the new configuration. Configurations with lower total potential are more likely to be accepted. The number of times that a particle is selected (i.e., number of sintering steps) per coagulating step are determined by the ratio of sintering to growth rates.

This method has been successful in providing insight in the formation of agglomerates. However, the possibility of extending it to include coagulation and condensation processes is limited by the weakness of the method in computing the physical time, [69] which is critical in the case of processes occurring at two different timescales like the case of the encapsulation process where coagulation and condensation occurs simultaneously but at different rates.

Instead, coagulation can be simulated through a more straightforward and physical Monte Carlo approach developed by Husar, [55] based on the work of Schaad [67] and Sutherland. [70] In these approaches, position and velocity space are omitted and only collisions are simulated, not in the physical sense but as a transition in a Markov chain. The following section illustrates the concept of Markov chains, and the use of it to simulate coagulation processes.

1.2. The Markov process

A Markov process is a chain of events occurring in sequence with the condition that the probability of each subsequent event in the chain is not influenced by prior events. [71] The

usual example is a drunk gentleman who begins a walk through a strange city. At each street corner that he reaches, he continues his walk by choosing completely at random one of the streets leading from the intersection. The history of this random walk is then a Markov chain, because his decision at any point is not influenced by where he has been.

Defining $S_i = \{x_1, x_2, \dots, x_p\}$ as the set of variables that fully describe the state of a system at time t_i , a Markov chain is defined as the finite set of states $M\{S_1, S_2, \dots, S_n\}$ through which the system evolves. At each of the discrete sequences of time t_i , the state S_i determines a set of conditional probabilities $q_{i1}, q_{i2}, \dots, q_{im}$, where q_{ij} is the probability that the system which is in state S_i , will be in state $S_{i+1} = S_{ij}$ at the $(i+1)^{\text{th}}$ time. In other words, q_{ij} is the probability of the transitions $S_i \rightarrow S_{ij}$. It is important to note that the probability of the transition depends only upon the current state S_i and is not affected by the previous history of the system. This is the characteristic *Markovian* property of a process. The state S_f is said to be absorbing if the system remains in this state with probability one. A state S_i is linked with the state S_j if there is a nonzero probability that the system reaches the state S_j in a finite number of steps. A Markov chain is terminating if each of the states S_i is linked to an absorbing state S_f .

In the example of the drunk gentleman in the strange city, the state or position of the gentleman is described by his $\{x, y\}$ coordinate in the city at every discrete time. The collection of corners where he has been defines the Markov chain. Because of the randomness of his choice at each intersection, the transition probability at each corner is $q_{ij} = q = 1/4$. However, if his choice were determined by his "feeling" of safety, affected for example by the luminosity of the street, or the number of people on the street, the transition probabilities would be different at each state (corner), i.e., $q_{ij} \neq q_{ij+1} \neq q_{im}$, but his choice would be still unaffected by the places where he has been.

Because of the nature of the problem, it might be possible to simulate a sample walk by playing a roulette wheel with 100 sections. Before every spin, the roulette wheel is divided in 4 sectors, each corresponding to a possible direction, such that each sector has a number of uniformly spaced sections proportional to the sense of safety that the drunk gentleman feels in that specific direction. Starting at a given position, the gentleman's most likely path and most likely position after certain time could be found by simulating a large number of histories, i.e., by running through the process many times. Another example of a Markov process is the coagulation of particles in aerosols.

2. Coagulation as a Markov process

Consider the evolution of an aerosol consisting initially of 5 particles of the same size (singlets). If the number of singlets in the system at time t_i is denoted by n_{i1} , the number of doublets (particles made of two singlets) by n_{i2} , and so on, the state of the aerosol at the i^{th} time step (collision) is defined by the set $S_i = \{n_{i1}, n_{i2}, \dots, n_{is}\}$. Imposing the condition of irreversible collisions, the transition from the initial state $S_1 = \{5, 0, 0, 0, 0\}$ to the final state $S_7 = \{0, 0, 0, 0, 1\}$ may proceed along any of the paths shown below.

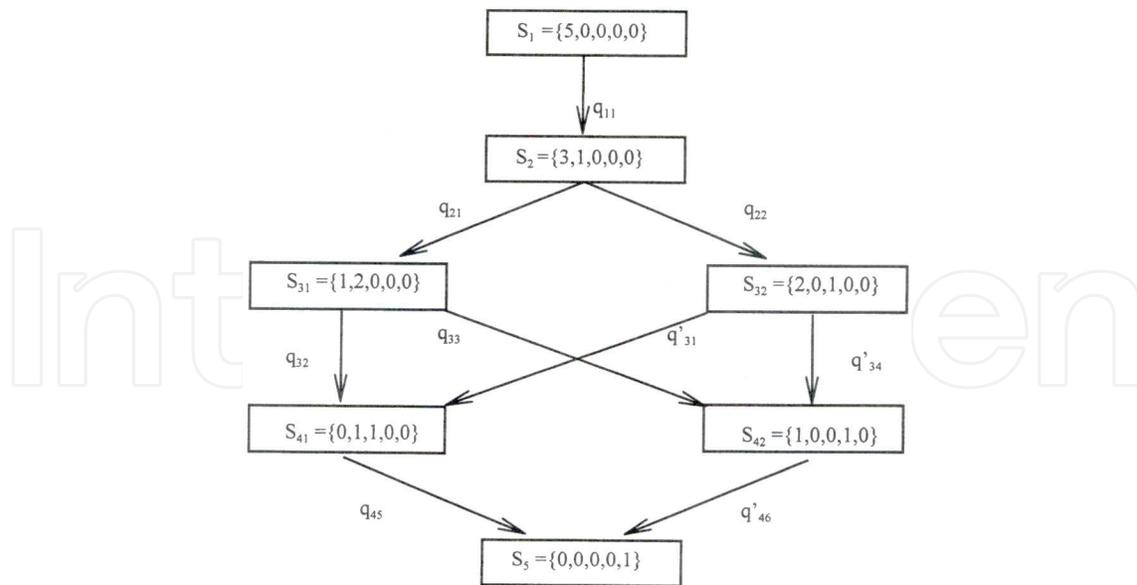


Figure 1. Coagulation as a Markov process. Possible path for the evolution of an aerosol consisting initially of 5 particles of the same size. (From Reference 55).

Every state S_i defines a set of possible types of collisions, each with a probability that is completely uninfluenced by the history of the aerosol. Therefore, the evolution of the aerosol is properly defined as a Markov process.

At time t_1 , only collisions among singlets are possible. Then $q_{11} = 0$ for $j \neq 1$. Therefore, the aerosol evolves to the state $S_2 = \{3, 1, 0, 0, 0\}$. At time t_2 , collisions type 1 (among singlets) and type 2 (singles with doublets) will happen with probability q_{21} and q_{22} , respectively. In general, at time t_i a collision type j (particles made of p -singlets with particles made of q -singlets) will occur with probability q_{ij} . The transition probability q_{ij} is proportional to the collision frequency $Z_{pq} = \beta_{pq} n_{ip} n_{iq}$. With the restriction:

$$\sum_j q_{ij} = 1 \tag{2}$$

then, the transition probability is given by:

$$q_{ij} = \frac{\beta_{pq} n_{ip} n_{iq}}{\sum \sum \beta_{pq} n_{ip} n_{iq}} \tag{3}$$

The objective of the Markov-Monte Carlo (Markov-MC) technique is to determine the most likely state at a given time t_i by simulating the transitions $S_1 \rightarrow S_i$, using random or statistical sampling of transitions. At every transition, the sampling of transitions can be done by, for example, aligning the collision probabilities q_{ij} on a 0–1 interval in an arbitrary order and then generating a random number R_n such that $0 < R_n < 1$. R_n identifies uniquely the types of collisions that have occurred, and consequently, the types of transitions that have happened.

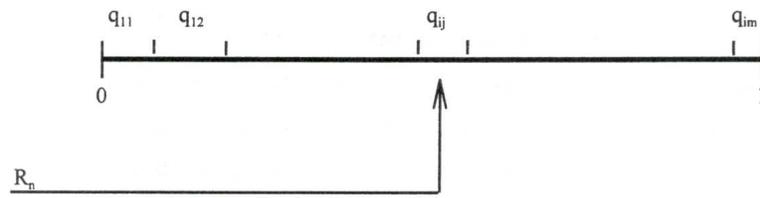


Figure 2. Arbitrary aligning of the transition probabilities in a starting line 0–1 such that a random number will identify uniquely the types of collisions that have happened. (From Reference 55)

Physically, the Markov-MC model simulates particle coagulation by reproducing the following algorithm. Time is divided into time steps. During each time step, the occurrence of every collision is registered and classified according to its type. When the information is normalized by Z , the total number of collisions that have occurred, it provides the probability of occurrence of each type of collision at the given time. This sampling process is Monte Carlo simulated by aligning the probabilities on a 0–1 line and generating Z random numbers, each one identifying a type of collision. In other words, by generating Z random numbers and correlating that number to the type of collision being statistically chosen, the collisions that have occurred during the time step are reproduced. Assuming that the time step is short enough, the change in the particle size distribution due to the occurrence of the Z collisions is negligible, and the change can be implemented at the end of the time step with minor error. By decreasing the time step, the error decreases. In the limit, the time step is chosen such that the particle size distribution is updated after every collision.

2.1. Computation of time

Computation of the physical time elapsed between events (i.e., the time step) has been a serious drawback of different Monte Carlo approaches. Most of the past studies have been performed where quantification of time can be avoided. However, when coagulation and condensation occur simultaneously, computation of time is a critical issue.

Schaad [67] proposed to use well-known solutions to calibrate the model with respect to time and then use those settings for the particular case of interest. Using the same approach, Kaplan and Gentry, [68] working with a Monte Carlo simulation to study agglomeration of nonspherical chainlike agglomerates in combustion-generated aerosols, choose the quantity $(N_o/N-1)$ as a measure by which to represent the rate of cluster growth, since coagulation theory for constant collision frequency functions predicts that this quantity is directly proportional to time. Clearly, these approaches offer reasonable approximations but are not exact.

With the Markov-chain version of the Monte Carlo technique (i.e., the Markov-MC method), the average time between events is computed through the collision frequency function Z_{ij} . Here, an event is defined as a collision among particles of any type. In a unit time and unit volume, there are Z_{11} collisions among particles of size 1 and 1 occurring simultaneously with Z_{ij} collisions among particles of size i and j . In total, there are $Z = \sum \sum Z_{ij}$ collisions per unit time and per unit volume.

This implies that in a unit volume, on average there is one collision every $1/Z$ unit times. Then the average elapsed time between events is given by: [55]

$$\delta t = \frac{1}{\sum_i \sum_j Z_{ij}} \quad (4)$$

If it were desired to simulate the process more rigorously the time increment should be selected from a *Poisson* distribution with the mean corresponding to Equation (4).

The coagulation equation is a time-dependent mass conservation expression for aerosols. Then a numerical solution of this nonlinear integro differential equation can be obtained by updating the particle size distribution function after every Markov-MC simulated collision and by computing the elapsed time through Equation (4). The implementation of this method on a computer code is straightforward. The following section discusses the specific computational aspects involved during its implementation.

3. Description of the code

3.1. Discretization of the particle size range

The Markov-MC algorithm starts by dividing into m arbitrary sections the range of particle sizes. A 1D vector (\mathbf{R}) is used to store this information. The finest grid can be obtained with *discrete sectionalizations* where section i corresponds to the size of a particle made of i mers. This type of sectionalization is appropriate for the study of the early stages of evolution of, for example, monodisperse aerosols. However, when the evolution of the aerosol is being watched for long periods of time, a sectionalization with a fine grid at smaller sizes together with a coarse grid at larger sizes is more appropriate. This can be done by using a *logarithmic sectionalization* ($i \propto \log R_i$). Some authors also use a *geometric sectionalization*, where the size of a section i corresponds to a particle with double the mass of the previous section. In other situations, the use of a simple *linear grid* is very convenient. Table 2 shows expressions for the different types of sectionalization.

Type of Sectionalization	Expression
Discrete	$R_i = i^{1/3} R_1$
Logarithmic	$R_i = 10^{1/p} R_1$
Geometric	$R_i = 2^{(i-1)} / R_1^3$
Linear	$R_i = (1 + i/p) R_1$

p parameter to adjust the number of section used according to the range covered

R_1 size of the elemental mer, or minimum particle size

Table 2. Expressions for the different types of sectionalization.

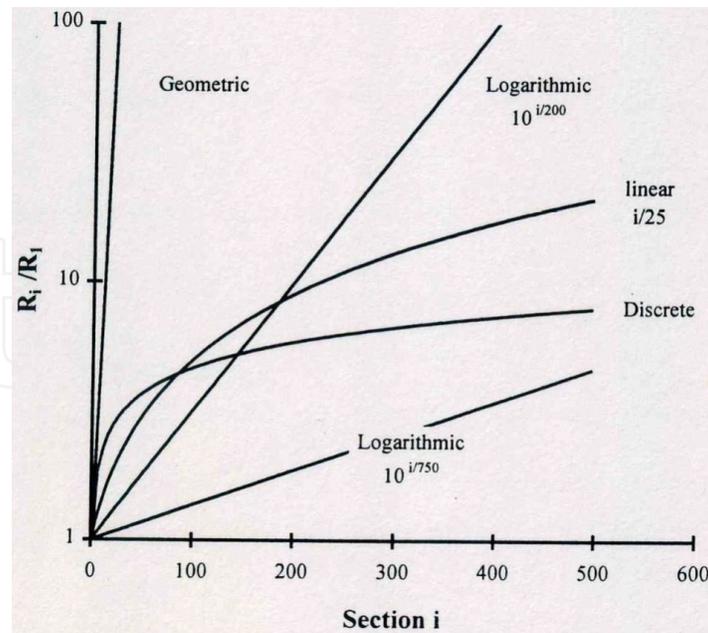


Figure 3. Types of sectionalization used in the Markov-MC model.

3.2. The concept of parcels

In actual combustion aerosols the particle concentration is typically greater than 1×10^{18} particles/m³. To directly simulate that many collisions would be impracticable, even with advanced supercomputers. To address this issue the concept of parcel was developed. A parcel is defined as a bundle of \tilde{n} identical particles, i.e., a bundle of particles having the same physicochemical characteristics (size, morphology, and chemical composition). Then, instead of simulating collisions among single particles, collisions among parcels are simulated. Consequently, at every step \tilde{n} collisions of the same type are Markov-MC simulated. For a given total number of particles per unit volume $\{N\}$, the scale factor ($f=N/\tilde{n}$) determines the total number of parcels available during the simulation and thus the statistics of the simulation itself. Greater accuracy is obtained with greater statistics and the statistics of the simulation increases with \tilde{n} . Physically, this approach is similar to simulate the process in smaller volumes. By simulating coagulation in, for example, 1 cm³ instead of 1 m³, the number of collisions that need to be simulated reduces by 10^6 , and the process is still accurately simulated.

3.3. Particle Concentration

Employing the concept of parcels, the discrete version of the particle size distribution (n_i) is stored in a 1D vector (array). n_i is the number of particles per unit volume of size:

$$R_i < r_i \leq R_{i+1} \tag{5}$$

where the actual particle size r_i is given by:

$$r_i = \left(\frac{M_i}{n_i} \frac{3}{4\rho\pi} \right)^{1/3} \quad (6)$$

and M_i is the total mass of the particles in section i . Notice that by using Equation (6), it has been assumed that all the particles within the section have the same size. Also notice that by allowing the size of the particles within a section to dynamically assume their mass mean value, instead of assuming that it remains constant, the total mass of the aerosol is conserved during the simulation of the evolution of the aerosol.

3.4. Aligning of the transition probabilities

In the Markov-MC method, the sampling of transitions between states is done by aligning the collision probabilities q_{ij} 's on a 0-1 interval in an arbitrary order. The implementation of this step is critical because in addition to align the transition probabilities, it should provide an easy and quick way to relate every transition probability q_{ij} with the respective type of collision (i-j).

A 1D vector \mathbf{P} was used to store the transition probabilities q_{ij} 's and a scheme consisting of a coarse followed by a fine search was adopted to relate q_{ij} to the type of collision that q_{ij} represents. q_{ij} is defined as the probability of occurrence of a collision between particles of size r_i and r_j . We let $r_i \leq r_j$ and define the probability \mathbf{P} vector as:

$$P_i = P_{i+1} + \sum_i^m q_{ij} \quad (7)$$

where $\sum q_{ij}$ is the probability that a collision involving a particle of size r_i , with another particle of the same or larger size occurs. P_i is the cumulative probability. A random number R_n between 0 and 1 will identify the collision that has occurred and a search process involving at most m steps (the coarse search) will identify i as the smaller particle involved in the collision. In the fine search, j , the second particle, will be identified through Equation (7) by solving for q_{ij} . This second search requires another m operation at most. In contrast, a plain algorithm where every q_{ij} is considered will require up to m^2 operations instead of $2m$ operations required by the present scheme. A more efficient alternative is desirable since most of the computational time is spent performing this step.

3.5. General algorithm

The overall algorithm to perform the Markov-MC simulation is described in the flow chart shown in Figure 4. Initially, the type of discretization, the starting particle size distribution n_i and the stopping condition (e.g., time for simulation) are specified as inputs to the program. Then, the computation of the collision coefficient β_{ij} for all the possible types of collisions is performed. This information is stored in an $m \times m$ matrix where m is the number of sections

used. Here it is assumed that the collision coefficient is constant within every section, which is a good assumption even with a very coarse sectionalization. This step needs to be performed only once during the entire simulation. Then a loop involving four steps is formed. In the first step, the transition probabilities are calculated and sorted in an array of size m representing the probability vector (\mathbf{P}). In the second, the time step is calculated according to Equation (4). In the third step, a random number (R_n) is generated, the corresponding location on \mathbf{P} is accessed, and the respective type of collision (i, j) is identified. During the fourth and last step, the particle size distribution function is updated, which involves the withdrawal of one particle from section i and j , and the addition of the new particle to the proper section (k), where $v_i + v_j = v_k$. In the actual code, a utility subroutine to check for mass conservation is used from time to time to search for possible computational errors. These steps are repeated until the stopping condition is reached.

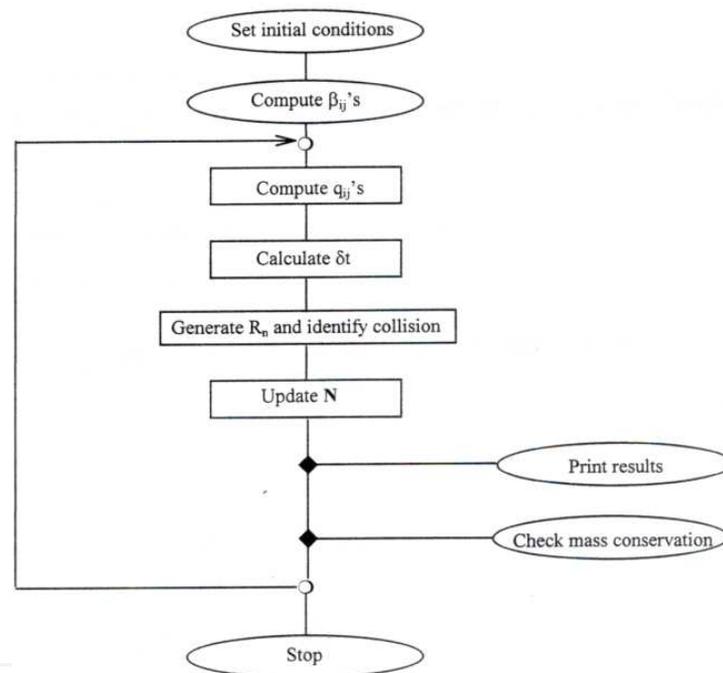


Figure 4. Flow chart illustrating the algorithm for the Monte Carlo simulation of coagulation processes.

4. Validation

Being a probabilistic approach, the calculation of the absolute accuracy of the Markov-MC method is generally impossible. However, its validity can be judged by, for example, comparing results with those obtained by other methods or a model problem whose solution is known accurately. Consequently, in this section the evolution of an initially monodisperse aerosol, with collision frequency function independent of particle size ($\beta_{ij} = \beta$), is Markov-MC simulated and compared to the corresponding analytical solution to test the accuracy of the

method. Later, the restriction $\beta = \text{constant}$ will be removed, and the Markov-MC results will be compared with other numerical results and experimental observations.

4.1. Evolution of an initially monodisperse aerosol with $\beta = \text{constant}$

4.1.1. The Smoluchowski's solution

One of the few analytical solutions to the GDE can be obtained for the case of aerosols subjected to the following restrictions:

- Continuum regime
- Collision frequency function β independent of particle size
- Initially monodisperse size distribution

With these assumptions and setting $r_i = r_j$, the collision frequency function reduces to:

$$\beta_{ij} = k_o = \frac{8KT}{3\mu} \quad (8)$$

Substituting this expression into Equation (1), the coagulation equation reduces to:

$$\frac{dn_k}{dt} = \frac{k_o}{2} \sum_{i+j=k} n_i n_j - k_o n_k \sum_{i=1}^m n_i \quad (9)$$

Summing over all values of k :

$$\frac{dN}{dt} = -\frac{k_o}{2} N^2 \quad (10)$$

Integrating once,

$$\frac{N}{N_o} = \frac{1}{1 + t / \tau_{coll}} \quad (11)$$

where N_o is the initial total number of particle per unit volume, and τ_{coll} is the characteristic coagulation time given by:

$$\tau_{coll} = \frac{2}{k_o N_o} = \frac{3\mu}{4KT N_o} \quad (12)$$

Physically, the characteristic time for coagulation is the time needed for an initial monodisperse aerosol to reduce the total particle concentration to half of its initial value. To obtain an expression for n_i we proceed inductively, i.e., first an expression for n_1 is obtained, then this result is used to obtain an expression for n_2 , and so on. Doing so, the general solution for the concentration of particles of size r_i is:

$$\frac{n_i}{N_o} = \frac{(t / \tau_{coll})^{i-1}}{(1 + t / \tau_{coll})^{i-1}} \quad (13)$$

Evolution of an initially monodisperse aerosol with constant collision frequency function were Markov-MC simulated. Results are shown in Figure 5 along with the analytical solution for N , n_1/N_o , n_2/N_o , and n_3/N_o . The agreement is excellent, demonstrating the validity of the Markov-MC model and the timescaling.

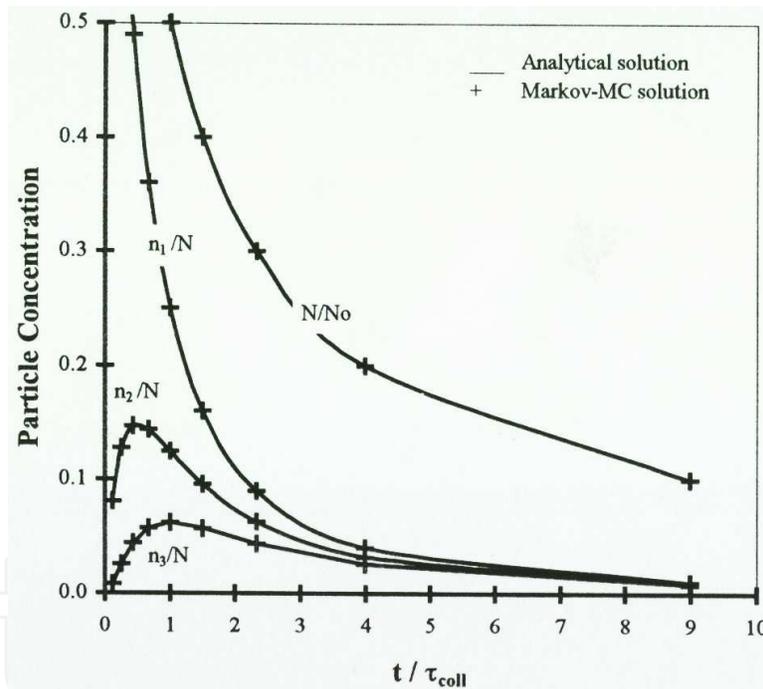


Figure 5. Markov-MC simulation of the evolution of an initially monodisperse aerosol with collision frequency function β independent of particle size. Good agreement between Markov-MC simulation results and the analytical solution are obtained.

Figure 6 shows the error in the total number of particles per unit volume at different stages of the evolution. The error in all cases is below 1% for $t/\tau_{coll} < 10$, showing a good agreement between the analytical solution and the Markov-MC simulation. The error arises from the computation of the time steps through Equation (4). This error can be minimized up to any degree by increasing the statistics of the simulation, i.e., by increasing the number of parcels \tilde{n} available during the simulation. However, simulations with higher \tilde{n} 's use greater amount

of computational resources. Figure 6 illustrates the effect of \tilde{n} on the error. It shows that the error is cumulative and increases exponentially with evolution.

The increase in error with time results because with every event (collision), the number of parcels available for simulation decreases by one. Consequently, the statistics of the simulation decreases with the number of events performed and, thus, the error in the computation of the time steps increases. The cumulative error in time can be maintained below a certain level without increasing the use of computational resources by dividing the entire simulation into several sequential runs. A run is stopped before the cumulative error in time is excessive. Before the next subsequent run, the number of parcels in each section is multiplied by 10 and the scale factor f is divided by 10, so that the total number of particles is conserved and the statistics of the simulation is increased. This practice allows the evolution of the aerosol to be carried out over very long periods of time, while minimizing the error in the computation of time and minimizing the use of computational resources.

To determine the appropriate initial number of parcel \tilde{n} to use during every run, several Markov MC simulations were conducted varying \tilde{n} . A balance between high statistics and use of computational resources was sought. Table 3 and Figure 6 show the results obtained. The values on Table 3 were obtained running the code on *Charney*, a NASA-Cray J932 machine, able to operate at 4.0 gigaflops. Flops is the number of floating point operations performed per second and a floating point operation is defined as any operation that manipulates numbers expressed in floating point notation. [72]

f	Number of Starting Parcel \tilde{n}	System CPU [s]	User CPU [s]
1x10 [12]	1x10 ⁶	1.4954	447.1
1x10 [13]	1x10 ⁵	0.3904	260.9
1x10 [14]	1x10 ⁴	0.2713	6.8
1x10 [15]	1x10 ³	0.2224	3.2

Table 3. Simulation of the evolution of an initially monodisperse aerosol with $N_0=1 \times 10^{18}$ [18] particles/m³.

In Table 3, the Central Processing Unit (CPU) time is the amount of system time required to run the job, i.e., the number of seconds devoted to system operations required for the program. The user CPU time is the amount of time that the program requires of the CPU.

Figure 6 and Table 3 show that working with *Charney*, the balance between accuracy and use of computational resources is obtained by starting with 1×10^6 parcels and increasing the resolution every 0.9×10^6 events. However, the required number of starting parcels depends on the machine used and the number of sections actively being used. For example, early stages of the evolution of the monodisperse aerosol require less computational operations than a fully developed one, since at the early stages most of the sections are empty, and the number of operations per event is proportional to the number of sections being actively used. On the other

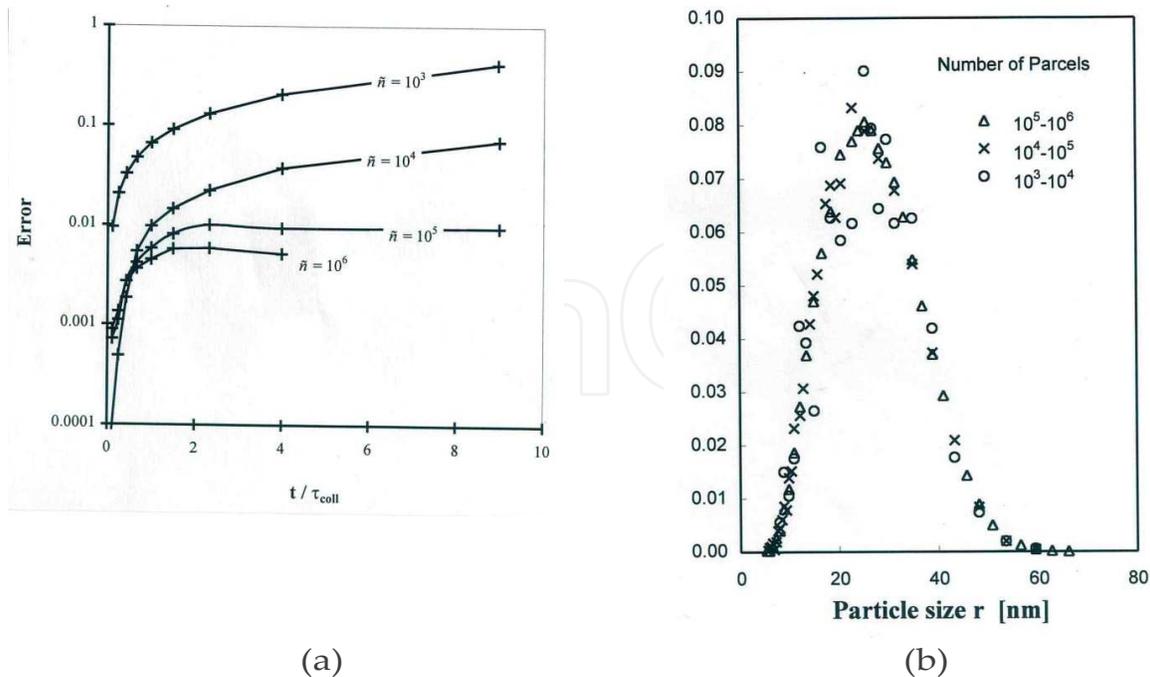


Figure 6. (a) Error of the Markov-MC method in computing the total number of particles per unit volume at different stages of the evolution of an initially monodisperse aerosol with constant collision frequency function β . (b) Comparative error in the evolution of an initially normal distributed aerosol as function of the initial number of parcels \bar{n} used.

hand, working with slower machines, resolution needs to be sacrificed to obtain results within a reasonable amount of time. For example, to obtain results within an hour, the number of starting parcels need to be reduced to 1×10^5 for *Megalon*, a local UNIX machine, while for a PC 110 MHz, the number of starting parcels need to be reduced to 1×10^4 .

4.1.2. Effect of the types of sectionalization and random number sequence

Several runs were made to observe the effect of different types of sectionalization on the Markov MC results. For the same conditions, discrete, geometric, linear, and logarithmic sectionalization was used. In all the cases, the results were the same. The possibility of using different types of sectionalization according to the problem being analyzed constitutes another advantage of the Markov-MC method over similar codes like the sectional method, which was designed to work on a logarithm type of sectionalization. A similar type of test was performed to see the effect of the number of sections being used on the evolution of a coagulating aerosol.

The evolution of an initially normal distributed aerosol was observed using a logarithmic type of sectionalization. Several runs were made, varying the number of section per decade being used. Figure 7.a shows the results obtained. This figure shows that the number of sections per decade can be reduced up to 18 and the evolution of the aerosol as predicted by the Markov-MC model is basically the same.

Similarly, the effect of the random number sequence being used was studied. C++, as any other programming language, provides a standard library function to generate pseudo random

numbers with a uniform distribution. The sequence of random numbers produced by this function is determined by a seed number. Varying this seed number, different sequences can be obtained. Figure 7.b shows the Markov-MC results obtained for the evolution of an aerosol for three different random number sequences. Figure 7.b shows that the Markov-MC results are independent of the sequence of random numbers used.

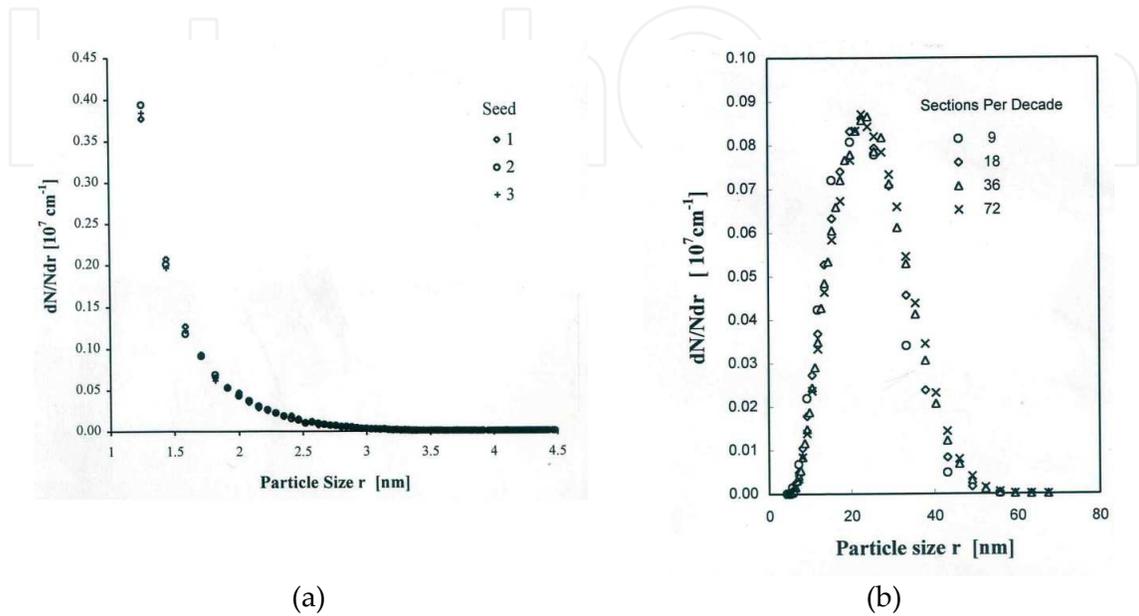


Figure 7. (a) Size distribution at an arbitrary time for an initially normal distributed aerosol as predicted by the Markov-MC model using a logarithmic type of sectionalization with different numbers of sections per decade. (b) Size distribution at $t/\tau_{\text{coll}} = 45.3$ ($\tau_{\text{coll}} = 49 \mu\text{s}$) for an initially monodisperse aerosol ($N_0 = 4.6 \times 10^{19}$ [19] particles/ m^3 , $r_1 = 1 \text{ nm}$) as predicted by Markov-MC method using three different random number sequences (seeds).

4.2. Comparison with numerical and experimental results

In general, the collision frequency function β is not a constant but depends on the sizes of the colliding particles and the nature of the aerosol. Under these conditions, there is not an exact analytical solution to the coagulation equation. Many numerical and experimental works have been performed to describe the behavior of coagulating aerosols. The main results are summarized here.

Later, the Markov-MC results obtained for an atmospheric aerosol with $\beta(r)$ are qualitatively compared with these numerical and experimental results.

Friedlander et al. [52] stated that the particle size distribution in a coagulating aerosol approaches an asymptotic self-preserving form after long periods of time, and appears to be independent of the initial size distribution. The self-preserving distribution for the case of $\beta = \text{constant}$ is exponential in form, whereas the self-preserving distribution for Brownian coagulation appears to be lognormal. Thus, for Brownian coagulating aerosols, the state of the aerosol at any time could be characterized by only two parameters, the mean particle size and the geometric standard deviation σ_g .

The theory was later supported by Hidy [73] who through numerical experiments concluded that the self-preserving form develops after about 3 dimensionless coagulation times ($3\tau_{\text{coll}}$) and that the shape of the asymptotic distribution varies with the Knudsen number (K_n) but not with the initial conditions. Later, Lee et al. [74] assumed a lognormal type of self-preserving distribution and found an approximate analytical solution for the coagulation equation by using the method of moments. The solution predicts an asymptotic value of $\sigma_g = 1.38$ in the free molecular regime. The solution also estimates the time to reach the asymptotic distribution in the range from 0 to $473\tau_{\text{coll}}$ depending on the initial conditions.

Matsoukas and Friedlander, [75] studying the dynamics of the formation of metal oxides (MgO and ZnO) in a flat flame where salts of magnesium and zinc were introduced in the form of a dry aerosol, measured the particle size distribution of samples collected at different heights above the flame. They observed that the distribution collapses into a single curve when plotted in normalized form and concluded that the normalized distribution is, to a good approximation, lognormal with a geometric standard deviation $\sigma_g \cong 1.4$. Similarly, Megaridis, [76] studying the morphology of the soot formed in a coannular ethane diffusion flame, obtained a value for $\sigma_g = 1.39\text{--}1.46$. Akhtar et al., [77] studying formation of TiO_2 by gas-phase synthesis, observed that the mean particle size evolves logarithmically while the geometric standard deviation moves very quickly toward $\sigma_g \cong 1.4$.

To validate the Markov-MC method, results for a coagulating atmospheric aerosol (i.e., water in air at room temperature) with $\beta(r)$ were obtained and compared with these numerical and experimental results. Initially, the aerosol consists of $N_0 = 1 \times 10^{18}$ [18] water particles/ m^3 of 1 nm in radii.

Figure 8 shows the evolution of the aerosol as predicted by the Markov-MC method. The results are presented in terms of nondimensional particle size distribution (n_i/N) as a function of the nondimensional time (t/τ_{coll}). The characteristic time for collision at the initial conditions is used to nondimensionalize time. Figure 8 shows that the particle size distribution gradually broadens out with time and evolves toward a self-preserving distribution that seems to be lognormal.

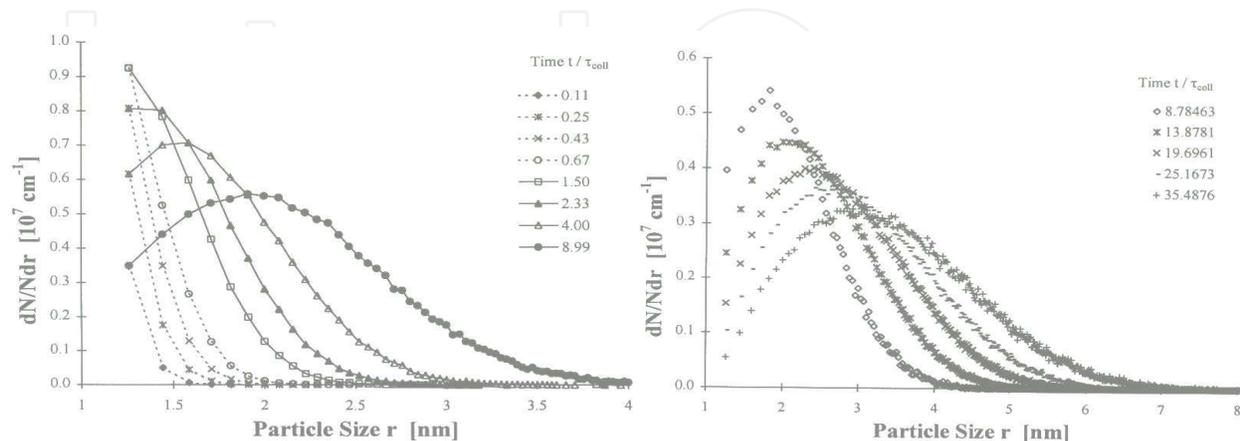


Figure 8. Size distribution of an initially monodisperse aerosol ($N_0 = 1 \times 10^{18}$ particles/ m^3 , $r_1 = 1$ nm, $\tau_{\text{coll}} = 2.25$ ms) as predicted by the Markov-MC method. The aerosol approaches a lognormal distribution after $\cong \tau_{\text{coll}}$.

To determine the type of self-preserving distribution predicted by the Markov-MC method, the obtained Markov-MC distributions were fitted with different exponential distribution functions. The goodness-of-fit was evaluated by the *chi-square test*, [64, 78] and it was found that among the exponential distributions, the lognormal distribution function produced the best fit to the data. The mean particle size r_m and the geometric standard deviation σ_g of the lognormal distribution were evaluated and Figure 9 shows the values of r_m and σ_g obtained as a function of time. This figure shows that the mean size r evolves logarithmically and the geometric standard deviation σ_g moves very quickly toward $\sigma_g \cong 1.45$, in agreement with other numerical and experimental observations.

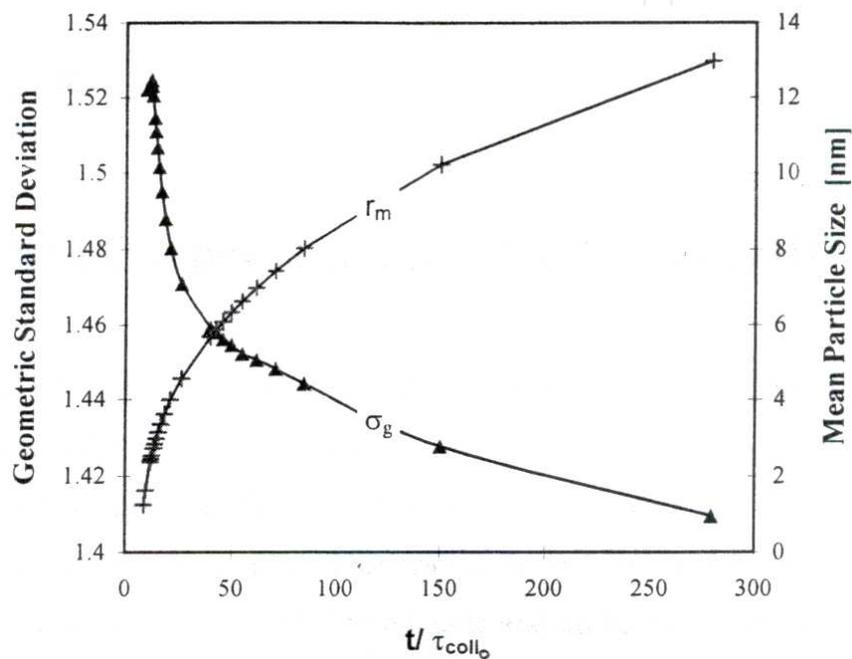


Figure 9. Evolution of the mean particle size r_m and the geometric standard deviation σ_g of the particle size distribution as predicted by the Markov-MC method, assuming that the distributions are lognormal.

5. Overview

A Monte Carlo method has been developed to simulate particle collisions in aerosols. Collisions are simulated as transitions in a Markov chain. The Markov-Monte Carlo (Markov-MC) method has been validated by comparing results with analytical and numerical results and with experimental observations. The method can handle any number of sections and any type of sectionalization. Conditions to obtain good accuracy with minimum use of computational resources were determined. The convenience, ease, directness, and expressiveness are some of the advantages of the method. However, the unique advantages of the Markov-MC method appear when condensation processes are added into the model. This is the subject of the next chapter.

Author details

Jose Ignacio Huertas

Address all correspondence to: jhuertas@itesm.mx

Tecnológico de Monterrey, Mexico

IntechOpen

IntechOpen

