This is a repository copy of *A Fast O(N2) Fragmentation Algorithm*.

White Rose Research Online URL for this paper:
https://eprints.whiterose.ac.uk/213878/

Version: Published Version

Article:

Rafikov, R.R., Silsbee, K. and Booth, R.A. orcid.org/0000-0002-0364-937X (2020) A Fast O(N2) Fragmentation Algorithm. The Astrophysical Journal: Supplement Series, 247 (2). 65. ISSN 0067-0049

https://doi.org/10.3847/1538-4365/ab7b71

CrossMark

A Fast $O(N^2)$ Fragmentation Algorithm

Roman R. Rafikov[1,2] , Kedron Silsbee[3] , and Richard A. Booth[4]

[1] Department of Applied Mathematics and Theoretical Physics, Centre for Mathematical Sciences, University of Cambridge, Wilberforce Road, Cambridge CB3 0WA, UK; rrr@damtp.cam.ac.uk
[2] Institute for Advanced Study, 1 Einstein Drive, Princeton NJ 08540, USA
[3] Max-Planck-Institut für Extraterrestrische Physik, D-85748 Garching, Germany
[4] Institute of Astronomy, University of Cambridge, Madingley Road, Cambridge CB3 0HA, UK

Abstract

Collisional fragmentation is a ubiquitous phenomenon arising in a variety of astrophysical systems, from asteroid belts to debris and protoplanetary disks. Numerical studies of fragmentation typically rely on discretizing the size distribution of colliding objects into a large number N of bins in mass space, usually logarithmically spaced. A standard approach for redistributing the debris produced in collisions into the corresponding mass bins results in $\mathcal{O}(N^3)$ calculation, which leads to significant computational overhead when N is large. Here, we formulate a more efficient explicit $\mathcal{O}(N^2)$ fragmentation algorithm, which works when the size spectrum of fragments produced in an individual collision has a self-similar shape with only a single characteristic mass scale (which can have arbitrary dependence on the energy and masses of colliding objects). Fragment size spectra used in existing fragmentation codes typically possess this property. We also show that our $\mathcal{O}(N^2)$ approach can be easily extended to work with non-self-similar fragment size distributions, for which we provide a worked example. This algorithm offers a substantial speedup of fragmentation calculations for large $N \gtrsim 10^2$, even over the implicit methods, making it an attractive tool for studying collisionally evolving systems.

Unified Astronomy Thesaurus concepts: Debris disks (363); Planetesimals (1259); Algorithms (1883)

1. Introduction

The issue of collisional fragmentation regularly arises in astrophysical problems where the masses of colliding objects— e.g., planetesimals or dust particles in protoplanetary disks— need to be followed. Examples include evolution of the asteroid belt (Durda & Dermott 1997), the Kuiper Belt (Davis & Farinella 1997; Kenyon & Bromley 2004), dust populations in debris disks (Kenyon & Bromley 2005; Krivov et al. 2008), and protoplanetary disks (Brauer et al. 2008; Birnstiel et al. 2010). The large number of objects in these applications makes it convenient to characterize the state of the system via the mass distribution (spectrum) $n(m)$, such that the number of objects in the mass interval $(m, m + dm)$ is $n(m)dm$. Pair-wise collisions cause mass to be exchanged between different parts of the mass space: a collision between objects m_1 and m_2 produces a number of fragments, channeling mass toward smaller objects. The total mass in the system of colliding objects is usually conserved in the process, although it may be lost in very energetic collisions when vaporization occurs; for simplicity, we will disregard the latter possibility. Also, when particles reach very small sizes, they could be removed from the system by other processes, such as Poynting–Robertson drag or radiation pressure.

In a continuous limit, the evolution of the mass spectrum due to fragmentation in pair-wise collisions is described by the following equation:

$$\frac{\partial n(m)}{\partial t} = \frac{1}{2} \int_0^\infty \int_0^\infty dm_1 dm_2 \; g(m|m_1, m_2)$$
$$\times \, R(m_1, m_2) n(m_1) n(m_2)$$
$$- n(m) \int_0^\infty dm_1 R(m, m_1) n(m_1). \quad (1)$$

Here, $R(m_1, m_2)$ is the rate coefficient for collisions between particles of masses m_1 (target) and m_2 (projectile), while $g(m|m_1, m_2)$ is the size (or mass) distribution of fragments produced in a single collision. It is defined such that the number of fragments in the mass interval $(m, m + dm)$ resulting in a collision between particles m_1 and m_2 is $g(m|m_1, m_2)dm$. The first term in Equation (1) is the source and represents the production of particles of mass m in collisions of particles with masses m_1 and m_2, with a factor of $1/2$ introduced to avoid double counting. Note that $g(m|m_1, m_2)$ can be identically zero, e.g., when $m_1 + m_2 < m$. The second term in Equation (1) describes the removal of particles in a given mass bin in disruptive collisions with other particles of all sizes. Note that, although we do not consider the possibility of particle coagulation in this work, our subsequent results are applicable also to systems, in which both fragmentation and coagulation operate.

In numerical applications, the mass coordinate is discretized into a large number N of bins (typically uniformly spaced in $\ln m$). The number of objects per ith bin is $n_i(t)$ and the vector $\boldsymbol{n} = \{n_i\}, \quad i = 1, \ldots, N$ fully characterizes the system. Equation (1) is then evolved in two steps. First, one chooses a pair of bins i and j, $i, j = 1, \ldots, N$ and computes the number of collisions between particles in these bins that occur in time Δt. Second, the fragments produced in collisions of each mass pair are distributed over the N bins according to their mass distribution (i.e., the function $g(m|m_1, m_2)$), which is described by the first term on the right-hand side of this equation. This procedure needs to be repeated for each i, j pair of colliding bins.

The first step requires $\mathcal{O}(N^2)$ operations in general, while the second takes $\mathcal{O}(N)$. As a result, the numerical cost of evolving Equation (1) scales as[5] $\mathcal{O}(N^3)$ per time step. This can be rather

---

[5] This numerical cost of the fragmentation calculation should be compared with the $\mathcal{O}(N^2)$ cost of the coagulation computation, which is less demanding.

challenging when N is very large, which is often needed to provide an accurate description of the collisional evolution of astrophysical systems spanning many orders of magnitude in mass.

The goal of this work is to demonstrate that the numerical cost can be reduced to $\mathcal{O}(N^2)$ for a certain class of fragment size distributions, which is rather common, and flexible enough to handle even more general models of collision outcomes. We describe this fragmentation model in Section 2, and the associated $\mathcal{O}(N^2)$ algorithm in Section 3. We then show how this model can be extended to approximate more general forms of the fragment size distribution (Section 4), and provide a numerical illustration in Section 5. We compare explicit and implicit methods for evolving fragmentation cascades in Section 6. Our results are discussed in Section 7.

2. Fragmentation Model

The outcome of a collision between two objects depends on a variety of factors. The primary ones are the masses m_1 and m_2 involved in a collision, the relative velocity of the colliding objects $v_{\rm rel}$, and the material properties of each object determined by the composition, structural characteristics (e.g., porosity), and size. The precise geometry of the collision (i.e., impact parameter for spherical objects) also plays an important role. In many studies, the rates of the collisions and collision outcomes are treated in an averaged sense, by convolving over the distributions of $v_{\rm rel}$, impact parameters, etc. In this work, to simplify the notation, we will explicitly spell out the dependence of collision outcomes only on the masses of the colliding objects, implicitly assuming that the collisional velocities (or characteristics of their distribution, such as the mean and dispersion) and other parameters depend on masses as well. Given that the relative velocities may also depend on time, the collision outcomes should typically be recalculated at every time step.

We characterize the size distribution of fragments forming in a collision of objects with mass m_1 and m_2 using a reasonably general model of a collision outcome. It covers the two most common possibilities: (1) the erosion in weakly energetic collisions, which results in one dominant post-collision remnant with the mass $m_{\rm rm}(m_1, m_2)$ and a continuous spectrum of small fragments; and (2) catastrophic disruption, when the large remnant no longer exists and only a continuous spectrum of fragments remains. This model has a form

$$g(m|m_1, m_2) = \epsilon\delta(m - m_{\rm rm}) + g_f(m|m_1, m_2), \quad (2)$$

where $\epsilon = 1$ in the case of erosion, while $\epsilon = 0$ in the case of catastrophic collisions. Here, $g_f(m|m_1, m_2)$ is a mass spectrum describing a continuous population of fragments formed in a collision; see Figure 1 for a schematic illustration. There are other possible outcomes of particle collisions—sticking, mass transfer, bouncing, etc. (Güttler et al. 2010; Windmark et al. 2012)—which we do not consider in this study.

A particular form of g_f explored in this work that allows one to reduce the computational cost of the fragmentation calculation to $\mathcal{O}(N^2)$ is the *self-similar* fragment mass spectrum:

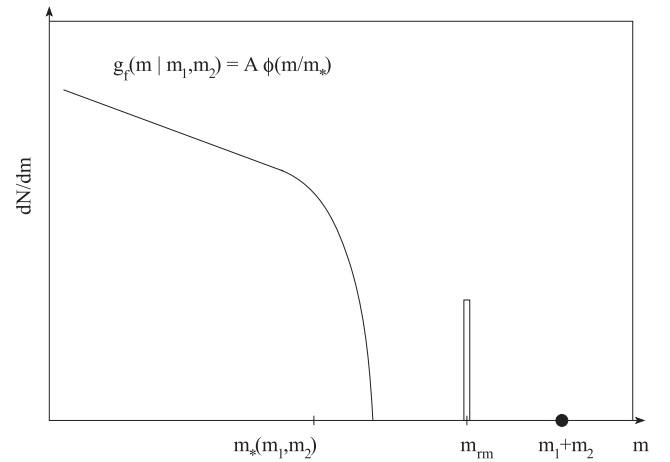$$g_f(m|m_1, m_2) = A\varphi\left(\frac{m}{m_*(m_1, m_2)}\right). \quad (3)$$

Figure 1. Schematic representation of an outcome of a collision (mass distribution of collision products) between the two objects with masses m_1 and m_2, represented by Equation (2). Collision products consist of: (1) a large remnant with mass $m_{\rm rm}$ (for $\epsilon = 1$); and (2) a continuous spectrum of small fragments $g_f(m|m_1, m_2)$, which in this case has a self-similar form (3). Various characteristic mass scales are indicated on the horizontal axis.

Here, φ is an arbitrary function that truncates at large masses m; m_* is the characteristic mass scale set by m_1, m_2, and the details of collision physics (i.e., collision energy); and A is the normalization of the spectrum. As we will show later in Section 3, the key feature of this fragment mass spectrum is that all information about the collision details, e.g., the time-dependent relative velocity of the colliding objects (which sets both the rate of collisions and the size distribution of fragments), etc., is absorbed in a single parameter—the mass scale m_*.

The value of A is set by mass conservation (in the absence of mass losses to vaporization)

$$m_1 + m_2 = \epsilon m_{\rm rm} + A\int_0^\infty m\varphi(m/m_*)dm, \quad (4)$$

so that

$$A = \frac{m_1 + m_2 - \epsilon m_{\rm rm}}{Im_*^2}, \quad I = \int_0^\infty z\varphi(z)dz. \quad (5)$$

Integration over the mass coordinate can be to extended to infinity since the function $\varphi(z)$ vanishes for large values of z.

Laboratory experiments suggest (Gault & Wedekind 1969; Hartmann 1969; Fujiwara et al. 1977; Blum & Münch 1993) that the fragment mass spectrum can often be described reasonably well by a power law in fragment mass m truncated above some largest fragment mass $m_* = m_{\rm lf}$:

$$g_f(m|m_1, m_2) = A\begin{cases} (m/m_{\rm lf})^\gamma, & m \leqslant m_{\rm lf}, \\ 0, & m > m_{\rm lf}. \end{cases} \quad (6)$$

For example, Fujiwara et al. (1977) found that the mass spectrum of fine fragments resulting in collisions of basaltic bodies can be well-described by a power-law dependence with index $\gamma = -1.8$. On the other hand, Blum & Münch (1993) found that $\gamma = -9/8$ in their experiments with $ZrCO_4$ aggregates. The spectrum (6) has the self-similar form (3) with

$$\varphi(z) = \varphi^{\rm pl}(z|\gamma) = \Theta(1 - z)z^\gamma, \quad (7)$$

where $z = m/m_{\rm lf}$ and $\Theta(z)$ is the Heavyside step function (superscript "pl" stands for "power law").

3. $O(N^2)$ Fragmentation Algorithm

We now demonstrate how the fragmentation calculation described by the Equation (1) can be turned into an $\mathcal{O}(N^2)$ problem, rather than $\mathcal{O}(N^3)$, for the fragment mass spectrum in the form (3). We will later show in Section 4 that this procedure can be generalized to cover even more complicated fragment mass spectra.

The basic idea behind this algorithm lies in the order in which different steps are performed. In the standard $\mathcal{O}(N^3)$ approach, for every pair of mass bins, the calculation of the collisional debris production is immediately followed by the redistribution step—assigning fragments to their corresponding mass bin. In our new method, the order is different: after computing debris production for each mass bin pair, we bin the outcomes (spectrum amplitudes A) according to their m_* and $m_{\rm rm}$, which is possible when $g_f(m|m_1, m_2)$ has a self-similar shape (3). Only after this procedure has been carried out for all pairs of bins do we perform the fragment distribution step. Both of these steps, performed sequentially, can be computed in $\mathcal{O}(N^2)$ operations, providing the desired speedup. We next describe this algorithm in detail.

Let us introduce two auxiliary N-dimensional vectors. One is $\boldsymbol{n}^{\rm rm} = \{n_i^{\rm rm}\}$ and is used to record the number of large remnant bodies resulting from erosive collisions in a fixed time interval that end up in the ith bin (i.e., with $m_{\rm rm}$ falling into this bin). Another vector is $\mathcal{A} = \{\mathcal{A}_i\}$—the sum of normalization factors A given by Equation (5) for all collisions that have the characteristic mass scale of their fragment size distributions m_* falling into the ith mass bin.

We can now describe our $\mathcal{O}(N^2)$ fragmentation algorithm step by step.

1. At the start of a new time step, we set $n_i^{\rm rm} = 0$, $\mathcal{A}_i = 0$, $i = 1, \ldots, N$.
2. We pick a particular mass bin $i = 1, \ldots, N$, which is an $\mathcal{O}(N)$ operation.
3. We first take care of the last term in the right-hand side of Equation (1), although the order is not important. We consider collisions of objects in the ith bin with objects in every $j = 1, \ldots, N$ bins in the system, calculating their rate $R(m_i, m_j)$ and the actual number of collisions in time dt:

$$dN_{ij}^{\rm coll} = R(m_i, m_j)n(m_i)n(m_j)dm_i dm_j dt, \quad (8)$$

where dm_i, dm_j are the width of the ith and jth mass bin, respectively. Note that $dN_{ij}^{\rm coll}$ can be a noninteger.

The total loss of particles from the ith bin is then

$$dn_i^- = \sum_{j=1}^{N} dN_{ij}^{\rm coll}. \quad (9)$$

Calculation of $dN_{ij}^{\rm coll}$ and dn_i^- for all combinations of i and j requires $\mathcal{O}(N^2)$ operations.

4. We then deal with the first term in the right-hand side of Equation (1) and consider the spectrum of fragments resulting in collisions between objects in ith and jth bins considered before. Standard fragmentation algorithms directly distribute these fragments for each i, j pair into

the relevant mass bins already at this step (another $\mathcal{O}(N)$ operation). This would make the algorithm scale as $O(N^3)$.

We proceed differently. Knowing the relative energy of collision, we compute the values of $m_{\rm rm}$, m_* and A for every pair of i and j. We then treat large remnants and small debris as follows.

Large remnant bodies: We find the index $k_{\rm rm}$ of the bin into which $m_{\rm rm}$ falls, and increase the value of $n_{k_{\rm rm}}^{\rm rm}$ by the number of remnant bodies produced in $dN_{ij}^{\rm coll}$ collisions:

$$n_{k_{\rm rm}}^{\rm rm} \rightarrow n_{k_{\rm rm}}^{\rm rm} + \epsilon \, dN_{ij}^{\rm coll} \frac{m_{\rm rm}}{m_{k_{\rm rm}}}. \quad (10)$$

The factor $m_{k_{\rm rm}}/m_{\rm rm}$ is introduced here to conserve mass: it accounts for the fact that the large remnant mass $m_{\rm rm}$ does not necessarily equal the central mass of the bin $m_{k_{\rm rm}}$.

Small fragments: We then take care of the continuous spectrum of smaller fragments. First, we determine the index k_* of the bin into which m_* falls. Since, again, in general $m_{k_*} \neq m_*$, we ensure mass conservation by adjusting A to a (slightly different) value A', such that

$$A' = A\left(\frac{m_*}{m_{k_*}}\right)^2. \quad (11)$$

This follows from the fact that the total mass of the self-similar fragment size spectrum with mass scale m_* is $Am_*^2 I$, where I is the integral defined in Equation (5).

We then update the k_*th component of the vector $\mathcal{A}$ as follows:

$$\mathcal{A}_{k_*} \rightarrow \mathcal{A}_{k_*} + dN_{ij}^{\rm coll} A'. \quad (12)$$

5. Operations in steps (2)–(4) are repeated for all pairs of i and j (avoiding double counting). This, in general, requires $\mathcal{O}(N^2)$ calculations, at the end of which vectors $n_i^{\rm rm}$ and $\mathcal{A}_i$ get fully updated.
6. Now we go through the final, redistribution, steps. We first update the number of objects n_i in each $i = 1, \ldots, N$ bins as follows:

$$n_i \rightarrow n_i + n_i^{\rm rm} + \sum_{j=1}^{N} \mathcal{A}_j \varphi\left(\frac{m_i}{m_j}\right) dm_i, \quad (13)$$

where dm_i is the width of ith mass bin. In other words, we add to each bin all the large remnants and small fragments that originally fell within its corresponding mass interval. This step again uses $\mathcal{O}(N^2)$ operations.

Finally, contributions (sinks) from step (3) are subtracted for all $i = 1, \ldots, N$ bins:

$$n_i \rightarrow n_i - dn_i^-, \quad (14)$$

adding $\mathcal{O}(N)$ additional operations.

7. Time is incremented by dt and steps (1)–(6) are repeated once again.

One can see that this algorithm indeed performs the fragmentation calculation using only $\mathcal{O}(N^2)$ operations per time step, and not $\mathcal{O}(N^3)$ as the conventional approach. This improvement can be achieved only when the collision outcome is described by the Equation (3), with the mass spectrum of

fragments being a self-similar function φ with a *single characteristic mass scale* m_*.

Indeed, if φ depended on, e.g., two mass scales, then the amplitude vector $\mathcal{A}$ would need to be replaced with the two-dimensional $N \times N$ amplitude array. In that case, the first part of the redistribution step (6) would have involved $\mathcal{O}(N^3)$ operations, since the summation in Equation (13) would need to be carried out over two indices. Similarly, methods based on implicit time integration are $\mathcal{O}(N^3)$; see Section 6. Nevertheless, in the following sections, we will show how our $\mathcal{O}(N^2)$ algorithm can be applied also to some more general collision outcomes than the one given by the Equation (3).

4. Generalizations of the Algorithm

The method presented in the previous section allows some straightforward generalizations that greatly extend its applicability. Such generalizations are possible when the spectrum of fragments can be represented or approximated using the self-similar components. We cover both cases below.

4.1. Superposition of Self-similar Mass Spectra

A rather straightforward extension of the algorithm outlined above is possible when the fragment mass spectrum can be represented as a sum of $L > 1$ self-similar mass distributions:

$$g_f(m|m_1, m_2) = \sum_{\eta=1}^{L} A_\eta \varphi_\eta \left(\frac{m}{m_\eta(m_1, m_2)} \right), \qquad (15)$$

where characteristic mass scales m_η are distinct (i.e., not multiples of each other). In this case, amplitudes A_η can no longer be found from Equation (5). Instead, they need to be specified independently, with the only constraint coming from the mass conservation:

$$m_1 + m_2 = \epsilon m_{\mathrm{rm}} + \sum_{\eta=1}^{L} A_\eta m_\eta^2 I_\eta, \qquad (16)$$

$$I_\eta = \int_0^\infty z \varphi_\eta(z) dz. \qquad (17)$$

For example, the full mass spectrum (2) can be viewed as a sum of two self-similar components: remnant spectrum $\epsilon \delta(m - m_{\mathrm{rm}})$ with the amplitude ϵ and mass scale m_{rm}, and the continuous self-similar spectrum of small fragments given by Equation (3).

The only difference from the procedure described in Section 3 is that instead of one amplitude vector $\mathcal{A}_i$ we would introduce now L such vectors and then repeat the steps (4)–(6) for all L individual self-similar contributions. The number of operations would scale as $\mathcal{O}(LN^2)$.

4.2. Piecewise Approximation of the Fragment Spectrum

Our algorithm can also be used when the spectrum of the fragments can be approximated in a piecewise fashion using a number of self-similar components. For example, almost any fragment spectrum can be represented as a series of $S > 1$

power-law segments (within certain mass intervals) of the form

$$g_f(m|m_1, m_2) \approx \sum_{s=1}^{S} \Psi_s(m),$$

$$\Psi_s(m) = A_s \begin{cases} 0, & m < m_s^{\mathrm{min}}, \\ \left(\dfrac{m_s^{\mathrm{max}}}{m} \right)^{\gamma_s}, & m_s^{\mathrm{min}} \leqslant m \leqslant m_s^{\mathrm{max}}, \\ 0, & m > m_s^{\mathrm{max}}, \end{cases} \qquad (18)$$

where γ_s is some average of $d \ln g_f(m) / d \ln m$ within the interval $m_s^{\mathrm{min}} \leqslant m \leqslant m_s^{\mathrm{max}}$. Fragment size distributions in the form of two (or more) broken power laws have been found in the collisional experiments of Fujiwara et al. (1977), Takagi et al. (1984), Davis & Ryan (1990). However, any reasonably smooth fragment mass spectrum can be approximated in this way, given a sufficiently large number of components (mass intervals) S.

Each of these components can be written as the difference of the two power law spectra φ^{pl} defined by Equation (6), namely

$$\Psi_s(m) = A_s \varphi^{\mathrm{pl}} \left(\frac{m}{m_s^{\mathrm{max}}} \ \middle| \ \gamma_s \right)$$

$$- A_s \left(\frac{m_s^{\mathrm{min}}}{m_s^{\mathrm{max}}} \right)^{\gamma} \varphi^{\mathrm{pl}} \left(\frac{m}{m_s^{\mathrm{min}}} \ \middle| \ \gamma_s \right). \qquad (19)$$

Combining Equations (18) and (19), we see that $g_f(m|m_1, m_2)$ ends up being approximated as a linear combination of $2S$ self-similar (power law) components, which reduces the problem to the one already considered in Section 4.1.

Note that one does not have to approximate g_f as the sum of only power-law segments $\Psi_s(m)$; other representations are possible too, as will be shown next.

5. Example Calculation: Piecewise Approximation of the Fragment Spectrum

To demonstrate the accuracy and speedup associated with using our $\mathcal{O}(N^2)$ algorithm, we now provide an example of applying it to treat collisional evolution with a non-self-similar fragment size distribution. We consider a fragment mass spectrum

$$g_f(m|m_1, m_2) = A \exp\left[-\left(\frac{m}{m_{*1}} \right)^3 \right] \left(\frac{m}{m_{*1}} \right)^\alpha$$

$$\times \left[1 + \left(\frac{m_{*2}}{m} \right)^2 \right]^{(\alpha-\beta)/2}, \qquad (20)$$

with normalization A, two mass scales, m_{*1} and m_{*2}, and two power law slopes α and β. This mass spectrum is exponentially truncated above m_{*1}. It behaves as a power law $m^{-\alpha}$ for $m_{*2} \lesssim m \lesssim m_{*1}$; however, the power-law slope smoothly changes to β for very small fragments, $m \lesssim m_{*2}$.

Very importantly, m_{*1}/m_{*2} is not a constant, but changes as the collision characteristics (e.g., masses m_1 and m_2) vary. This makes $g_f(m|m_1, m_2)$ given by Equation (20) non-self-similar, which is illustrated in Figure 2. There, we show how the shape of $g_f(m|m_1, m_2)$ (multiplied by m^2) evolves as m_{*1} and m_{*2} vary, implying lack of self-similarity. Each of the curves is normalized such that the total mass in fragments is always
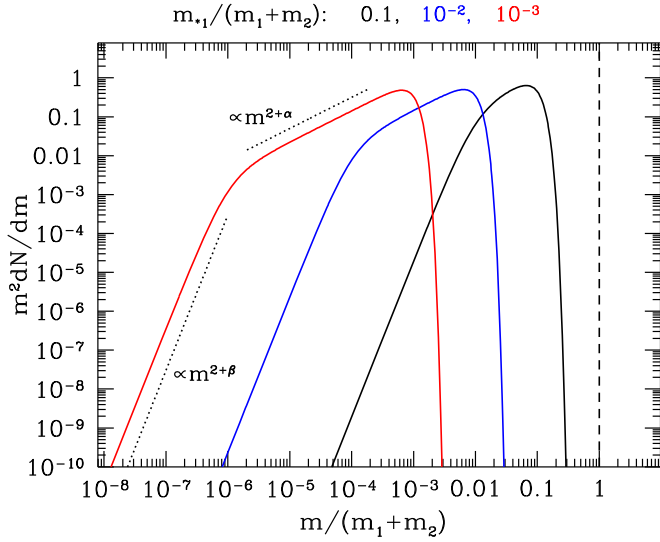
Figure 2. Fragment mass spectrum given by Equation (20), shown for three different values of $m_{*1} = 0.1, 10^{-2}, 10^{-3}$ (with corresponding $m_{*1} = 10^{-2}, 10^{-4}, 10^{-6}$). We used $\alpha = -1.2$ and $\beta = 2$ in this illustration. The color scheme is illustrated at the top. Note the non-self-similar shape of the spectrum revealed by its evolution as m_{*1} changes.

$m_1 + m_2 - \epsilon m_{rm}$ (although in this figure, we set $\epsilon = 0$ for simplicity).

We can approximate the mass spectrum (20) as a superposition of three self-similar components as follows:

$$
g_f(m|m_1, m_2) \approx A \exp\left[-\left(\frac{m}{m_{*1}}\right)^3\right]\left(\frac{m}{m_{*1}}\right)^\alpha
$$
$$
- A_- \varphi^{\mathrm{pl}}\left(\frac{m}{m_{*2}}\ \middle|\ \alpha\right) + A_+ \varphi^{\mathrm{pl}}\left(\frac{m}{m_{*2}}\ \middle|\ \beta\right), \tag{21}
$$

each of them featuring only one mass scale. This decomposition is illustrated in Figure 3. The first term (green dotted curve) is an exponentially truncated power law extending all the way down to very small fragment sizes; it is designed to fit the original spectrum (20) for $m \gtrsim m_{*2}$. The second term (blue dashed line) is a power law with the same slope α as in the first term, sharply truncated above m_{*2}. Its amplitude A_- is chosen so that it fully offsets the first term below m_{*2} (note that it enters with the negative sign). Finally, the last component (dashed magenta line) is another power law sharply truncated at m_{*2}, with the slope β and amplitude A_+ chosen such that this term matches the behavior of the spectrum (20) for $m \lesssim m_{*2}$.

We now carry out two fragmentation calculations. One uses fragment mass spectrum (20) without approximations; because of its non-self-similar shape, this calculation employs the standard $\mathcal{O}(N^3)$ fragmentation algorithm. The second calculation uses an approximation (21), allowing us to use our $\mathcal{O}(N^2)$ algorithm as described in Section 4.2. Both of them use explicit time stepping (see Section 6 for comparison with implicit calculations). In both cases, we evolve the system using Euler's method. The time step is chosen so that the number of particles in one bin will not change by more than 10% in any one time step (with an allowance for bins with a small number of particles in them). We then compare the outcomes of the two calculations, as well as the numerical costs involved.
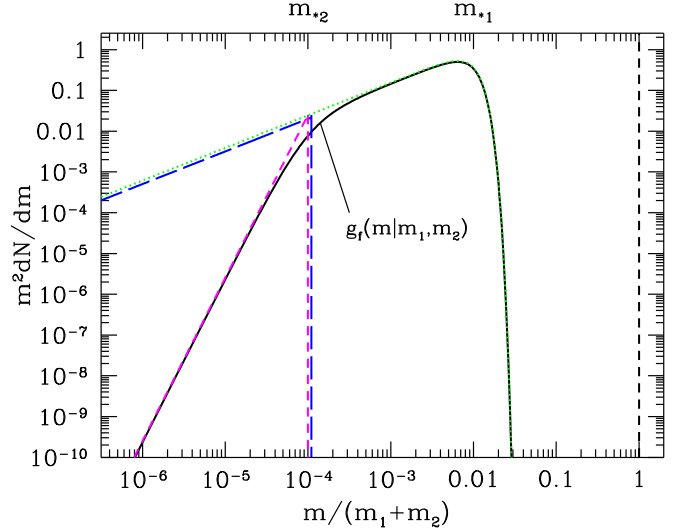

Figure 3. Decomposition of the fragment mass spectrum (20) into three different self-similar components, as shown in Equation (21). See text for details.

In both cases, we assume that m_{*1} is given by

$$
m_{*1} = \frac{m_1}{10^2}\left(\frac{m_1}{m_2}\right)^{1/2}, \tag{22}
$$

where $m_1 \geqslant m_2$. We also choose

$$
m_{*2} = \frac{m_{*1}^2}{m_1 + m_2}, \tag{23}
$$

so that when $m_{*1}/(m_1 + m_2)$ goes down (e.g., for more energetic collisions), there is a larger range in $\ln m$, for which $g_f(m|m_1, m_2) \propto m^\alpha$ (i.e., more small fragments get formed). We use $\alpha = -1.5$ and $\beta = 2.5$ in this calculation. At time, $t = 0$ all mass in the system is in objects with the same mass $m = 1$ (monodisperse initial condition) occupying a single mass bin. The mass interval that we cover extends from $m = 10^{-16}$ to $m = 1$. For this problem, we assume that fragments falling below the lower-mass end get removed from the system, representing the removal of small particles from the fragmentation cascade in debris disks by noncollisional processes such as Poynting–Robertson drag, radiation pressure on dust grains, etc.; this does not affect the operation of our algorithm in any way. However, strict mass conservation can easily be enforced for problems in which the removal of small debris is not expected (e.g., by assuming that the smallest bin includes all particles with size smaller than the minimum covered by the mass grid).

For simplicity, we assume that collisions lead to fragmentation only if $m_2/m_1 \geqslant 10^{-2}$. Also, we assume that no largest remnant remains, i.e., only the continuous spectrum of small fragments results in a fragmentation event. The collision rate is proportional to the geometric cross section of the two colliding bodies, assuming all objects to be spheres of the same density. This setup is similar to that in Dohnanyi (1969) and Tanaka et al. (1996), except for the non-self-similar shape of the fragment spectrum.

Results of the two calculations are shown in Figure 4. There, we plot the mass distributions dN/dm (multiplied by m^2 and normalized to the initial mass and particle number) at different
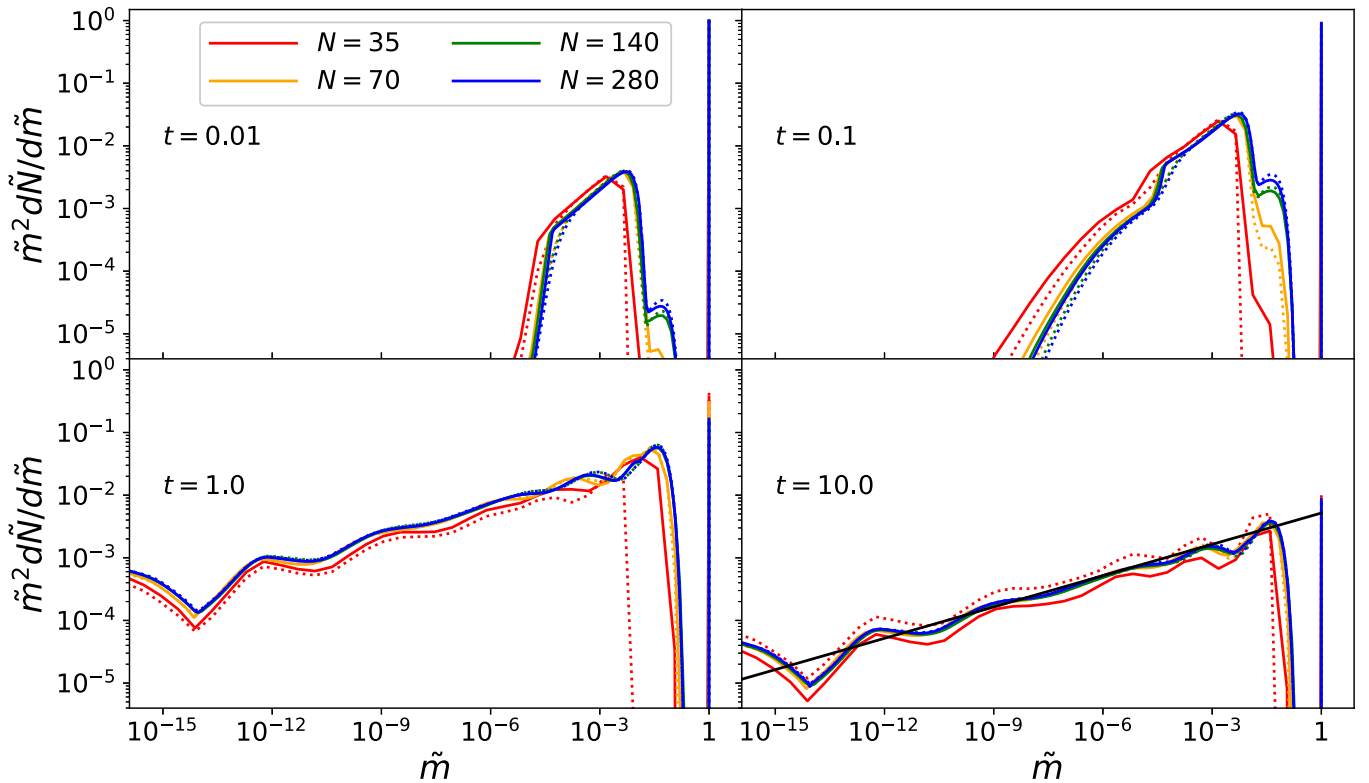
Figure 4. Mass spectrum (multiplied by m^2 to provide a proxy for a cumulative mass distribution in the system) at four different times indicated on the panels, expressed in units of collisional timescale for the initial mass bin. We use $\tilde{m}$ and $\tilde{N}$ for m and N, normalized by the initial mass ($m = 1$) and initial number of objects in the system. Different-colored curves correspond to different numbers of bins in the simulation, labeled in panel (a). Solid lines correspond to the curves from the $\mathcal{O}(N^2)$ algorithm, and dotted lines to those from the $\mathcal{O}(N^3)$ algorithm. In the bottom right panel, we show in black the line of slope $1/6$ (consistent with Dohnanyi 1969), which provides a good overall fit to the fragment distribution. The superimposed wavy structure is discussed in the text.

times during the calculation, for the two different algorithms run with different numbers of mass bins. Time, labeled on the panels, is in terms of the initial timescale for collisions between bodies in the initial mass bin. The height of the bin at $m = 1$ shows the current number of particles in the initial mass bin normalized to the initial number of objects in the system.

We first discuss the general features of the collisional evolution in this calculation. Early on, at $t = 0.01$, the mass spectrum closely mirrors that of the assumed fragment size distribution (20). This is to be expected because, at that time, the number of objects with $m < 1$ is small enough (total mass in this part of the spectrum is $\lesssim 1\%$) for their mutual collisions to be rare. On the other hand, collisions between these fragment and the numerous large $m = 1$ objects do occur, which explains a bump[6] appearing above $m = 10^{-2}$. This bump becomes more pronounced at $t = 0.1$, and fully morphs with the continuous mass spectrum by $t = 1$.

By $t = 0.1$, the shape of the distribution of fragments starts to evolve away from the single-collision spectrum (20) at small m. By $t = 1$, the fragment mass distribution attains a steady-state form, which can be viewed as a power law with superimposed wavy structure. The slope of this power law is close to $1/6$ (shown in black in the bottom right panel), in agreement with the results of Dohnanyi (1969) and Tanaka et al. (1996).

---

[6] According to Equation (22), collisions with smaller fragments result in larger m_{*1}. Because of our assumption of no fragmentation when $m_2/m_1 < 10^{-2}$, the largest possible m_{*1} is $0.1m_1$. This explains the gap between the initial mass bin at $m = 1$ and the continuous spectrum of fragments, which persists at all times.

The wiggles on top of this power law are caused by the boundary condition at the low-mass end; see Campo Bagatin et al. (1994) for a discussion of this effect. Beyond $t = 1$, only the normalization of the size distribution changes, steadily decaying in time because of mass lost to particles smaller than our smallest mass bin, while its overall shape stays the same. The height of the $m = 1$ bin goes down as well; it drops by two orders of magnitude by $t = 10$, signaling substantial erosion of the initial population of objects.

5.1. Comparison of the $\mathcal{O}(N^2)$ and $\mathcal{O}(N^3)$ Algorithms

We now compare the performance of the $\mathcal{O}(N^2)$ algorithm and the full $\mathcal{O}(N^3)$ calculation. We first note that, for small numbers of bins ($N = 35$), there are substantial differences between the results of the two calculations at all times. However, as the number of bins increases, the results converge, with two algorithms agreeing with each other quite well already for $N = 70$. Minor differences remain, especially at early times (when there is still a strong sensitivity to the shape of the input fragment spectrum), as even in the limit of an infinite number of bins, the fragment mass distributions are slightly different near m_{*2} (see Figure 3). This can be seen, for example, near $m = 10^{-5} - 10^{-4}$ (right around m_{*2} for collisions of two $m = 1$ objects) at $t = 0.01$ for $N = 280$. Nevertheless, at late times, these differences are largely wiped out. Thus, we can conclude that already with 5–10 mass bins per decade, our $\mathcal{O}(N^2)$ algorithm is able to reproduce the fine details of the collisional evolution even for a non-self-similar fragment size spectrum.
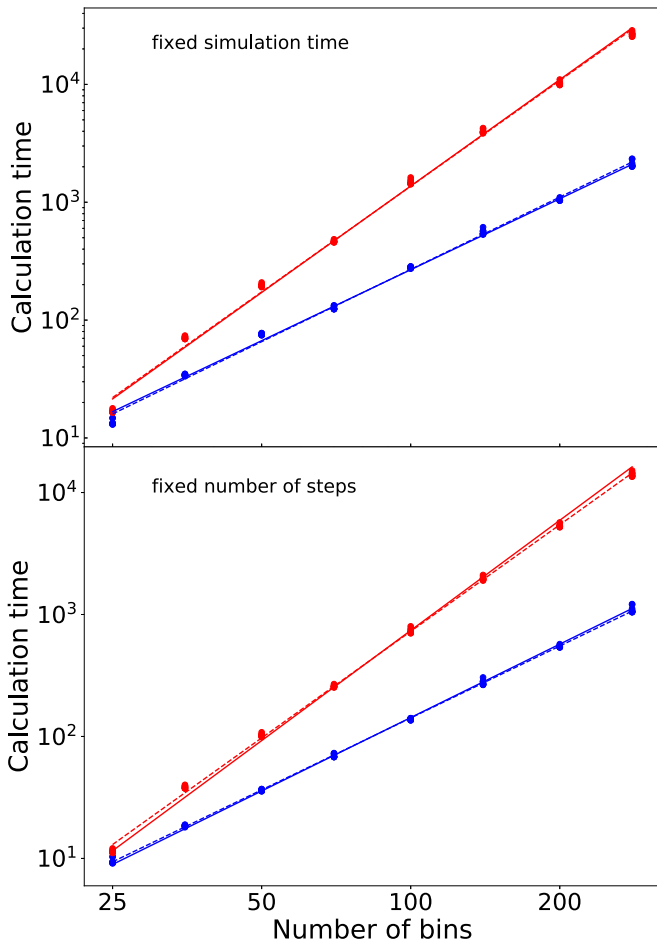
Figure 5. Run time as a function of the number of bins, expressed in arbitrary units. Data points in the top panel show the amount of time needed to reach a simulation time of $t = 1$ in units of initial collision times. Data points in the bottom panel show the time needed to run for 400 steps. Blue points and curves are for the $\mathcal{O}(N^2)$ algorithm, and red ones are for the $\mathcal{O}(N^3)$ algorithm. Solid lines are the best-fit power law to all the points, assuming the slopes to be exactly 2 and 3. Dotted lines are the best-fit power laws determined without fixing the slope. See text for more details.

Turning now to the computational cost of each algorithm, in Figure 5 we show the wall clock time to run them as a function of the number of bins N. We use an adaptive time step, so the number of time steps required to evolve the distribution for a given simulation time is slightly different between the two algorithms, and for different numbers of bins. For that reason, we plot both the amount of time required to reach a fixed simulation time ($t = 1$, upper panel), as well as the time required to execute 400 steps of the run (lower panel).

There are slight variations in the run time even for the exact same parameters, presumably caused by the evolving state of the computing hardware. For this reason, for each algorithm and number of bins, we run the calculation 10 times; hence the multiple points shown in Figure 5 for each number of bins. We then calculate the best-fit lines through the points for each algorithm, assuming $T \propto N^2$ and N^3 scalings, correspondingly. These are the solid lines in the figure. We also calculate the best-fit lines without fixing their slopes, which are shown as the dotted lines. These slopes turn out to be 2.04 and 2.98 in the top panel, and 1.96 and 2.90 in the bottom panel, in good agreement with the theoretical expectations.

6. Implicit versus Explicit Time Stepping

A number of studies have used implicit time integration methods to evolve the coagulation-fragmentation equations (Brauer et al. 2008; Birnstiel et al. 2010; Garaud et al. 2013; Booth et al. 2018). The advantage of these methods has traditionally been that they allow much longer time steps to be used in the integration, leading to a faster time to solution despite the increased complexity of the method. However, these studies did not make use of the $\mathcal{O}(N^2)$ fragmentation algorithm, which can only be used with explicit time stepping.

The difference between explicit and implicit methods can be summarized as follows. Master Equation (1) written in a discretized form in mass space has the form of a system of equations

$$\frac{\partial n_i}{\partial t} = f_i(t, \boldsymbol{n}), \tag{24}$$

where, as before, $\boldsymbol{n} = \{n_i\}$ and vector $\boldsymbol{f} = \{f_i\}$ stands for the expressions in the right-hand side of Equation (1) written out for each i. When evolving this system, explicit updates by δt in time take the form

$$n_i(t + \delta t) = n_i(t) + f_i(t, \boldsymbol{n}(t))\delta t, \tag{25}$$

whereas implicit updates reduce to solving the system of equations

$$n_i(t + \delta t) = n_i(t) + f_i(t + \delta t, \boldsymbol{n}(t + \delta t))\delta t \tag{26}$$

for $n_i(t + \delta t)$. Writing $F(n_i^*) = n_i^* - n_i(t) - f_i(t + \delta t, \boldsymbol{n}^*)\delta t$, the solution $F(n_i(t + \delta t)) = 0$ can be found via Newton–Raphson iteration, in which the estimate n_i^* is updated via

$$n_i^* \rightarrow n_i^* - \left[\hat{I} - \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{n}}\delta t\right]^{-1} F(n_i^*), \tag{27}$$

where $\hat{I}$ is the identity matrix.

The appearance of the Jacobian, $\partial \boldsymbol{f}/\partial \boldsymbol{n}$, and matrix inversion in the above equation prevents implicit methods from benefiting from the $\mathcal{O}(N^2)$ computation of the fragmentation rate because both the Jacobian computation and the matrix inversion in Equation (27) require $\mathcal{O}(N^3)$ operations to compute. The $\mathcal{O}(N^3)$ complexity of the Jacobian calculation can be understood from Equation (12), as $\partial \mathcal{A}/\partial \boldsymbol{n}$ is an $N \times N$ matrix.

To compare the efficiency of implicit $\mathcal{O}(N^3)$ and explicit $\mathcal{O}(N^2)$ schemes, we use a simplified version of the problem presented in Section 5.1. We take the fragment spectrum to have a self-similar form

$$g_f(m|m_1, m_2) = A \exp\left[-\left(\frac{m}{m_{*1}}\right)^3\right]\left(\frac{m}{m_{*1}}\right)^\alpha, \tag{28}$$

with m_{*1} defined by Equation (22) as before. In this case, to integrate these equations, we choose two third-order methods: the explicit third-order Runge–Kutta method of Gottlieb & Shu (1998), and the implicit third-order Rosenbrock method of Rang & Angermann (2005) used by Booth et al. (2018). Both of these methods provide an embedded error estimates, which are used to adapt the time step to ensure that the relative error is below 1%. For the implicit scheme, we use the traditional $\mathcal{O}(N^3)$ fragmentation algorithm, while the $\mathcal{O}(N^2)$ scheme is used with the explicit time integration scheme.
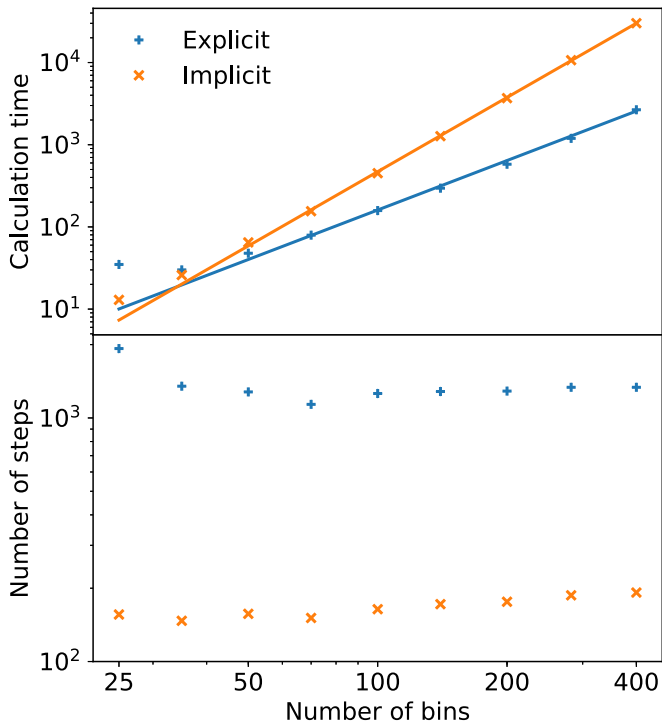
Figure 6. Comparison of the time required (top), shown in arbitrary units, and number of steps taken (bottom) to evolve the system for 10 initial collisional timescales for the third-order implicit (orange) and explicit (blue) methods. The lines show best fits assuming slopes of 2 and 3.

The time taken and number of steps required by the schemes to integrate the fragmentation equations to $t = 10$ in units of initial collisional timescales are shown in Figure 6. While the implicit method requires a factor of 7–9 fewer steps than the explicit scheme (bottom panel), the extra cost of the $\mathcal{O}(N^3)$ algorithm outweighs this for problems with more than about 50 cells (top panel). It is worth noting that the implicit method is more competitive for problems where the mass distribution is close to a steady state. Evidence of this is provided by considering the amount of time spent in the example shown in Figure 6 integrating from $t = 0$ to $t = 1$ (during which time the shape of the mass distribution changes considerably), compared with the time spent integrating from $t = 1$ to $t = 10$ (when only the normalization changes). The explicit method uses similar numbers of steps in each of the two time intervals, while the implicit method requires approximately 10 times fewer steps for the latter time interval in which the distribution is in a quasi-steady state. Tests on problems including both coagulation and fragmentation lead to similar conclusions.

7. Discussion

The main result of this work is the $\mathcal{O}(N^2)$ algorithm for numerical treatment of fragmentation in collisional systems, described in Section 3. The main condition necessary for this approach to work is that the continuous size distribution of fragments resulting in an individual collision $g_f(m|m_1, m_2)$ depends on a single mass scale m_*. This algorithm is insensitive to the details of the actual dependence of the different characteristics of the fragment mass spectrum —m_{rm}, m_*, and A, see Equations (2)–(3)—on the energy and masses of objects involved in a collision.

Self-similarity of $g_f(m|m_1, m_2)$ is not a highly demanding requirement, since the majority of numerical studies of fragmentation in astrophysical systems use such self-similar size distributions of fragments anyway, typically in the form of a truncated power law (6); see Greenberg et al. (1978), Kenyon & Luu (1999), Löhne et al. (2008), Brauer et al. (2008), and Birnstiel et al. (2010). On the other hand, implementation of this algorithm allows substantial gains in computational efficiency, significantly reducing the time consumed by fragmentation simulations with large number of mass bins, $N \gtrsim 10^2$; see Section 5.1.

Moreover, as we showed in Sections 4–5, the $\mathcal{O}(N^2)$ algorithm can be applied even when the fragment size spectrum is not a simple self-similar function with a single mass scale; one just needs to approximate the non-self-similar fragment size distribution using several self-similar components. A practical example shown in Section 5 demonstrates that the differences between a calculation carried with this algorithm and the direct "exact" $\mathcal{O}(N^3)$ calculation, which takes much longer, are very minor (at the level of several per cent or less) in systems that have evolved for longer than their characteristic collisional timescale. At early times, the level of agreement is dictated mainly by the accuracy with which the original complicated spectrum of fragments is approximated by the self-similar components.

There is a reason why approximating even rather complicated non-self-similar fragment size distributions (e.g., measured in some experiments, Fujiwara et al. 1977) with a simple self-similar shape works in practice. The characteristics of the steady-state collisional cascades are known to be rather insensitive to the input fragment size spectrum $g_f(m|m_1, m_2)$. For example, Tanaka et al. (1996) has shown that, as long as $g_f(m|m_1, m_2)$ is self-similar with $m_* \propto m_1$, the slope of the steady-state cascade should be sensitive only to the scaling of the collision rate with the masses of objects involved in a collision, but not to the actual form of $g_f(m|m_1, m_2)$. Similarly, O'Brien & Greenberg (2003) have shown that the slope of the collisional cascade depends on the mass scaling of the energy necessary to disrupt an object, but not on the power law of the fragment size spectrum (as long as $m_* \propto m_1$). By abandoning the assumption $m_* \propto m_1$, Belyaev & Rafikov (2011) were able to demonstrate the sensitivity of the steady-state cascade to the shape of the fragment size spectrum; however, the variation was found to be very weak (logarithmic). This is one of the reasons why, on long time intervals, after several collisional timescales have passed and the system settled into a steady-state cascade, our $\mathcal{O}(N^2)$ algorithm performs just as well as the exact calculation; see Figure 4(c) and (d).

Fragmentation algorithms documented in the literature (e.g., Greenberg et al. 1978; Kenyon & Luu 1999; Brauer et al. 2008; Löhne et al. 2008; Windmark et al. 2012; Garaud et al. 2013, etc.) redistribute the debris produced in collisions in the direct manner as described in Section 1 and are thus $\mathcal{O}(N^3)$. To the best of our knowledge, Booth et al. (2018) is the only other study mentioning the possibility of constructing $\mathcal{O}(N^2)$ algorithm for self-similar fragment size distributions, albeit without providing details. Our present study is intended partly to fill this gap.

A number of studies invoke implicit time integration (Brauer et al. 2008; Birnstiel et al. 2010; Garaud et al. 2013), which has an $\mathcal{O}(N^3)$ complexity due to the Jacobian calculation. The benefit of these schemes has traditionally been that the larger

time steps they allow outweigh the additional cost in solving the linear system, which is only a factor of ~2 when already using an $\mathcal{O}(N^3)$ fragmentation algorithm. Because our $\mathcal{O}(N^2)$ algorithm is more than one order of magnitude faster than an implicit method per step already for $N = 100$, this means that the time step for implicit methods must be smaller by a similar factor to remain competitive. Such a reduction in the number of steps was reported by Brauer et al. (2008) for a problem including both grain growth and radial drift, but this required computing both the coagulation and radial drift implicitly. Without computing radial drift implicitly, the explicit $\mathcal{O}(N^2)$ approach is substantially faster, as demonstrated in Figure 6. The simplicity of our $\mathcal{O}(N^2)$ algorithm also makes it easier to implement and parallelize, as well as using less memory than fully implicit methods (as the entire Jacobian need not be stored). This makes our algorithm more attractive for complex problems.

Some studies of coagulation/fragmentation in astrophysical systems characterize the constituent objects not only by their mass, but also by additional properties determining the collision outcome—e.g., particle porosity (Okuzumi et al. 2009), elemental abundances (Stammler et al. 2017), or charge (Akimkin et al. 2020). This would not affect the performance of our $\mathcal{O}(N^2)$ algorithm as long as the size distribution of fragments maintains the self-similar form (3). If only the mass scale m_*, but not the other characteristics of $g_f(m|m_1, m_2)$, start to depend on these additional properties, nothing should change in the logic of applying our algorithm in mass space.

Recently, simulations of dust dynamics in protoplanetary disks started including evolution of the dust size distribution due to coagulation/fragmentation *spatially resolved* in multiple dimensions (Drazkowska et al. 2019; Li et al. 2019). As this is done using the existing $\mathcal{O}(N^3)$ framework of Birnstiel et al. (2010), there is an associated computational overhead that scales steeply with the number of mass bins used to characterize dust size distribution. Use of our $\mathcal{O}(N^2)$ fragmentation algorithm could substantially reduce the computational cost of such calculations, making this tool an attractive option for future (multidimensional) studies of the dust evolution in disks around young stars.

ORCID iDs

Roman R. Rafikov ● https://orcid.org/0000-0002-0012-1609
Kedron Silsbee ● https://orcid.org/0000-0003-1572-0505
Richard A. Booth ● https://orcid.org/0000-0002-0364-937X

References

Akimkin, V. V., Ivlev, A. V., & Caselli, P. 2020, ApJ, 889, 64
Belyaev, M. A., & Rafikov, R. R. 2011, Icar, 214, 179
Birnstiel, T., Dullemond, C. P., & Brauer, F. 2010, A&A, 513, A79
Blum, J., & Münch, M. 1993, Icar, 106, 151
Booth, R. A., Meru, F., Lee, M. H., & Clarke, C. J. 2018, MNRAS, 475, 167
Brauer, F., Dullemond, C. P., & Henning, T. 2008, A&A, 480, 859
Campo Bagatin, A., Cellino, A., Davis, D. R., Farinella, P., & Paolicchi, P. 1994, P&SS, 42, 1079
Davis, D. R., & Farinella, P. 1997, Icar, 125, 50
Davis, D. R., & Ryan, E. V. 1990, Icar, 83, 156
Dohnanyi, J. S. 1969, JGR, 74, 2531
Drazkowska, J., Li, S., Birnstiel, T., Stammler, S. M., & Li, H. 2019, ApJ, 885, 91
Durda, D. D., & Dermott, S. F. 1997, Icar, 130, 140
Fujiwara, A., Kamimoto, G., & Tsukamoto, A. 1977, Icar, 31, 277
Garaud, P., Meru, F., Galvagni, M., & Olczak, C. 2013, ApJ, 764, 146
Gault, D. E., & Wedekind, J. A. 1969, JGR, 74, 6780
Gottlieb, S., & Shu, C. W. 1998, MaCom, 67, 73
Greenberg, R., Wacker, J. F., Hartmann, W. K., & Chapman, C. R. 1978, Icar, 35, 1
Güttler, C., Blum, J., Zsom, A., Ormel, C. W., & Dullemond, C. P. 2010, A&A, 513, A56
Hartmann, W. K. 1969, Icar, 10, 201
Kenyon, S. J., & Bromley, B. C. 2004, AJ, 128, 1916
Kenyon, S. J., & Bromley, B. C. 2005, AJ, 130, 269
Kenyon, S. J., & Luu, J. X. 1999, AJ, 118, 1101
Krivov, A. V., Müller, S., Löhne, T., & Mutschke, H. 2008, ApJ, 687, 608
Li, Y.-P., Li, H., Ricci, L., et al. 2019, ApJ, 878, 39
Löhne, T., Krivov, A. V., & Rodmann, J. 2008, ApJ, 673, 1123
O'Brien, D. P., & Greenberg, R. 2003, Icar, 164, 334
Okuzumi, S., Tanaka, H., & Sakagami, M.-A. 2009, ApJ, 707, 1247
Rang, J., & Angermann, L. 2005, BIT Numerical Mathematics, 45, 761
Stammler, S. M., Birnstiel, T., Panić, O., Dullemond, C. P., & Dominik, C. 2017, A&A, 600, A140
Takagi, Y., Mizutani, H., & Kawakami, S.-I. 1984, Icar, 59, 462
Tanaka, H., Inaba, S., & Nakazawa, K. 1996, Icar, 123, 450
Windmark, F., Birnstiel, T., Güttler, C., et al. 2012, A&A, 540, A73