Morphology-dependent random binary fragmentation of in silico fractal-like agglomerates

Supplementary Material

Y. Drossinos,1,* A. D. Melas,2,† M. Kostoglou,3,‡ and L. Isella4,§

1European Commission, Joint Research Centre, I-21027 Ispra (VA), Italy
2Aerosol & Particle Technology Laboratory, CPERI/CERTH, 57001 Thessaloniki, Greece
3Department, of Chemistry, Aristotle University, 54124 Thessaloniki, Greece
4European Commission, DG Trade, 1040 Bruxelles, Belgium

(Dated: September 3, 2019)

* yannis.drossinos@ec.europa.eu; Corresponding author
† amelas@cperi.certh.gr
‡ kostoglou@chem.auth.gr
§ lorenzo.isella@ec.europa.eu
Appendix A: Fragmentation conservation laws

We first show that the mass conservation law, eq. (3) in the main text, for the discrete fragment size distribution is a direct consequence of the first conservation law [expected number of fragments, eq. (2)]. The derivation is based on splitting the sum into two parts, redefining a summation index, and then using the symmetry property of the fragment size distribution, $p_{ij} = p_{(j-i)j}$. Specifically,

\begin{align}
\sum_{i=1}^{j-1} i p_{ij} &= \sum_{i=1}^{(j-1)/2} i p_{ij} + \sum_{i=(j-1)/2}^{j-1} i p_{ij} \quad \text{(decompose the sum)} \quad (A1) \\
&= \sum_{i=1}^{(j-1)/2} i p_{ij} + \sum_{i=1}^{(j+1)/2} (j-i) p_{(j-i)j} \quad \text{(change summation index, second sum)} \quad (A2) \\
&= \sum_{i=1}^{(j-1)/2} i p_{ij} + \sum_{i=1}^{(j-1)/2} (j-i) p_{(j-i)j} + \sum_{i=(j-1)/2}^{(j+1)/2} (j-i) p_{(j-i)j} \quad \text{(split second sum)} \quad (A3) \\
&= \sum_{i=1}^{(j-1)/2} i p_{ij} + \sum_{i=1}^{(j-1)/2} (j-i) p_{(j-i)j} + \sum_{i=(j-1)/2}^{(j+1)/2} (j-i) p_{(j-i)j} \quad \text{(use \(p_{ij}\) symmetry property)} \quad (A4) \\
&= j \sum_{i=1}^{(j-1)/2} p_{ij} + \sum_{i=(j-1)/2}^{(j+1)/2} (j-i) p_{ij}. \quad (A5)
\end{align}

A similar decomposition of the first conservation law leads to

\begin{align}
\sum_{i=1}^{(j-1)/2} p_{ij} &= 1 - \frac{1}{2} \sum_{i=(j-1)/2}^{(j+1)/2} p_{ij}. \quad (A6)
\end{align}

Substitution of eq. (A6) into eq. (A5) gives

\begin{align}
\sum_{i=1}^{j-1} i p_{ij} &= j + \sum_{i=(j-1)/2}^{(j+1)/2} \left(\frac{j}{2} - i\right) p_{ij} = j. \quad (A7)
\end{align}

The second sum in eq. (A7) is zero because the factor $j/2 - i$ changes sign with respect to $j/2$, whereas the distribution $p_{ij}$ is symmetric. Specifically for an odd number of monomers in the cluster $j = 2n + 1$ the sum becomes

\begin{align}
\sum_{i=n}^{n+1} (n + \frac{1}{2} - i) p_{i(2n+1)} &= \frac{1}{2} p_{n(2n+1)} - \frac{1}{2} p_{(n+1)(2n+1)} = 0, \quad (A8)
\end{align}
since \( p_{(n+1)(2n+1)} = p_{n(2n+1)} \). For \( j = 2n \) even, the relevant fragment sizes are \( n - (1/2) \) and \( n + (1/2) \). Since they are non-integers the distribution functions \( p_{(n-(1/2))(2n)} \) and \( p_{(n+(1/2))(2n)} \) are zero as fragments do not have non-integer number of monomers (alternatively, if half-integer fragments are considered, the previous argument for the sum being zero due to the symmetry properties of \( p_{ij} \) still holds ensuring that the sum is zero even for an even number of monomers in the initial cluster). Hence, for the discrete fragment distribution, the first conservation law (expected number of fragments) along with the symmetry property of the fragment size distribution imply the second conservation law (mass conservation).

The derivation applies mutatis mutandis to the continuous distribution \( p(x, y; d_f) \). Specifically, the mass conservation law becomes

\[
\int_{1}^{y-1} dx \, x \, p(x, y) = y + \int_{(y-1)/2}^{(y+1)/2} dx \left( \frac{y}{2} - x \right) p(x, y) = y. \tag{A9}
\]

The integral on the RHS of eq. (A9) vanishes because the function \((y/2 - x)p(x, y)\) is an odd function with respect to \(y/2\). Therefore, as in the case of the discrete fragment size distribution, only one conservation law is independent.

Appendix B: Cluster generation and description

The tunable cluster-cluster agglomeration algorithm [1, 2] used to generate the synthetic agglomerates is hierarchical in that after the generation of the initial building blocks, the clusters combine pairwise. A new cluster is generated from two pre-existing clusters by first choosing randomly a sticking point and a sticking angle: the two clusters stick at that point with the appropriate orientation. The two random choices ensure that the resulting agglomerate is unique. Then, one of the initial clusters is rotated by randomly choosing the three Euler angles. A distance condition on the rotated cluster, which ensures that the resulting cluster would satisfy the scaling law, is checked. If satisfied, the code checks whether monomers overlap: if they do not, the new agglomerate is accepted [3].

Each pairwise binding defines a generation. We considered two types of initial block units: dimers (\( N_{\text{init}} = 2 \)) and a collection of \( k \)-mers randomly chosen to have between six and eight monomers, \( N_{\text{init}} = 6, 7, 8 \). The number of monomers in a cluster of generation \( n \) is \( N_{\text{init}} \times 2^n \). The random choice of initial building blocks of a varying number of monomers gives clusters containing a range of monomer numbers, centered about a mode (the most
frequently occurring number of monomers in a set of clusters) that depends on the number of monomers in the initial building blocks (and the generation number). We tested two slightly different generation algorithms because we noted that random fragmentation of clusters generated with dimers only as the initial blocks tended to produce fragments containing a “magic” number of monomers (usually multiples of 2). Hence, fragmentation is a severe test of the cluster-generation algorithm. All the clusters considered in this work were generated with initial blocks of varying number of monomers.

TABLE I. Total number, size range (the minimum and maximum number of monomers in the set of \( n \)-th generation clusters), and mode (the most frequently occurring number of monomers in the set of \( n \)-th generation clusters) of the synthetic agglomerates \((k_f = 1.3)\).

| \(d_f\) | Gen 4 | Gen 5 | Gen 6 | Gen 7 | Total number |
|--------|-------|-------|-------|-------|--------------|
|        | Size range | Size range | Size range | Size range | |
|        | Mode | Mode | Mode | Mode | |
| 1.6    | 200,000 | 100,000 | 50,000 | 25,000 | 375,000 |
|        | 82 to 109 | 173 to 210 | 355 to 410 | 729 to 806 | |
|        | 96 | 192 | 385 | 769 | |
| 1.8    | 400,000 | 200,000 | 100,000 | 50,000 | 750,000 |
|        | 83 to 110 | 173 to 213 | 358 to 409 | 730 to 802 | |
|        | 96 | 192 | 384 | 770 | |
| 2.1    | 200,000 | 100,000 | 50,000 | 25,000 | 375,000 |
|        | 83 to 109 | 172 to 211 | 358 to 411 | 726 to 807 | |
|        | 96 | 192 | 384 | 767 | |

We generate clusters with fixed prefactor \((k_f = 1.3)\) and variable fractal dimension \(d_f = 1.6, 1.8, 2.1\). The clusters we analysed belonged to generations \(n = 4\) to \(n = 7\), their number varying from 200,000 (400,000) to 25,000 (50,000) for \(d_f = 1.6, 2.1\) \((d_f = 1.8)\). At every generation the number of clusters decreases by 2. The number of monomers per cluster varied from approximately 80 to 800, depending on the generation number. The number of clusters considered, their size range, and mode are shown in table I.
Appendix C: Discrete fragment size distribution

The discrete fragment size distribution is the beta distribution

\[ p_{ij} \equiv 2g_{ij} = 2 \frac{[i(j-i)]^{\beta(d_f,j)}}{\sum_{i=1}^{j-1}[i(j-i)]^{\beta(d_f,j)}} , \quad i = 1, \ldots, j - 1. \quad \text{(C1)} \]

The exponent \( \beta(d_f,j) \) is determined by requiring that

\[ g_{1j} = \frac{(j-1)^{\beta(d_f,j)}}{\sum_{i=1}^{j-1}[i(j-i)]^{\beta(d_f,j)}} \quad \text{(C2)} \]

equal the probability of monomer occurrence as determined from the empirical distribution, \textit{i.e.}, from the normalized histogram of fragment sizes. In fact, as in the case of the continuous distribution, eq. (C2) was compared to the numerically determined probability of monomer occurrence, as obtained from the fitting eq. (8) in the main text. The highly non-linear equation was solved numerically. The resulting exponent \( \beta(d_f,j) \) was fitted to the non-linear eq. (C3a), eq. (10a) in the main text and reproduced here for completeness,

\[ \beta(d_f,y) = a(d_f) + b(d_f) j^c(d_f). \quad \text{(C3a)} \]

to obtain the coefficients

\[ a(d_f) = -0.56 d_f - 0.25; \quad b(d_f) = 4.70 d_f + 2.47; \]
\[ c(d_f) = -0.76 d_f + 0.29 \quad \text{for} \quad d_f \in [1.6, 2.1]. \quad \text{(C3b)} \]

Figure 1 compares the calculated exponent of the beta distribution for the discrete \( (\beta(d_f,j)) \) and continuous \( (\beta(d_f,y)) \) fragment size probability distributions. The slight differences arise from the normalization condition: a discrete sum versus a definite integral.

[1] Thouy R. and Jullien R., \textit{J. Phys. I}, \textbf{6} (1996) 1365
[2] Filippov A. V., Zurita M. and Rosner D. E., \textit{J. Colloid Interface Sci.}, \textbf{229} (2000) 261
[3] Melas A. D., Isella L., Konstandopoulos, A. G. and Drossinos, Y., \textit{J. Colloid Interface Sci.}, \textbf{417} (2014) 27
FIG. 1. Exponent of the discrete $\beta(d_f, j)$ and continuous $\beta(d_f, y)$ beta fragment size distributions as a function of the initial agglomerate size ($j$ or $y$), parametrized by the fractal dimension (fractal prefactor $k_f = 1.3$). Discrete distribution (filled symbols), eqs. (C3); continuous distribution (open symbols), eqs. (10), main text.