Mater. Res. Soc. Symp. Proc. Vol. 1688 © 2014 Materials Research Society
DOI: 10.1557/opl.2014.870

An Analytic Toolbox for Simulated Filament Networks

Ronald J. Pandolfi[1], Lauren Edwards[1] and Linda S. Hirst[1]

[1]Dept. of Physics, School of Natural Sciences, University of California, Merced, California 95343, USA

ABSTRACT

Semi-flexible polymer networks generate a diverse family of structures. The network generating behaviors of specific semi-flexible biological filaments are well known (i.e. F-actin, microtubules, DNA etc.), however recent developments in tunable synthetic filaments extend the range of accessible structures. A similarly tunable model was developed using the molecular dynamics platform NAMD to provide a guide for generating synthetic filament networks. Structural characteristics of simulated networks may be quantitatively examined using connectivity analysis, radial pair distribution functions and scaling analysis. These methods provide a basis to calculate morphological properties, including mesh size, packing order, network connectivity, avg. cluster size, filaments per bundle, and space-filling dimensionality. An analytic toolset for describing the structure of filament networks is thus provided by detailing these methods.

INTRODUCTION

Semi-flexible polymer filaments are important to biological systems. For example, the cellular cytoskeleton is formed from F-actin, microtubule, and intermediate filaments cross-linked to form a network [1–3]. Recent experimental work has achieved the synthesis of semi-flexible polymers that can model the properties of these biological materials, but also have tunable properties [4]. This opens a new path for generating tunable hydrogels, applications for which have been discussed for drug delivery [5–7], tissue engineering [8], and mesoscale templating [2]. Such applications will benefit from the unique structure and rheology of these light yet rigid gels [9].

A diverse family of network-based materials can be generated from semi-flexible polymers, with structures distinct from those generated by flexible Gaussian polymers. In particular semi-flexible filaments have been demonstrated to generate branching networks of bundles – a structure unique to such systems [2,3,4]. Computational models for semi-flexible biological filaments (i.e. F-actin [10,11]) have been recently reported and agree well with experimental results.

While the basic solution behavior of biological filaments is fairly well known, a broader understanding of their self-assembly properties is needed to provide a generalized model that supports development of tunable synthetic polymers. Particularly in the formation of out-of-equilibrium structures such as the network of bundles. Our group's recent work has provided such a model describing semi-flexible polymer network formation [12] with generalized parameters allowing a range of structures to be generated (see Figure 1). In this paper we

describe a set of analytical techniques to quantitatively characterize the filamentous structures generated by coarse grained molecular dynamics simulation. These techniques provide a basis for the analysis of filament network structure through simulation.

In this paper we discuss three techniques that expose structural characteristics: the Radial-pair Distribution Function (RDF), Scaling analysis (Fractal dimension), and Percolation analysis. Properties which can be measured with these techniques include: network mesh size, filament packing structure, network connectivity, average cluster size, number of filaments per bundle, and space-filling dimensionality.

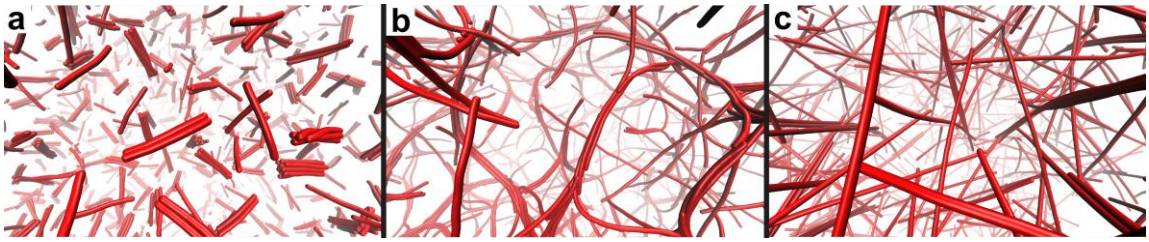


Figure 1. A molecular dynamics simulation snapshot of a system of isolated bundles (**a**), a branching network (**b**) and a non-branching network (**c**). All have a minimum filament spacing of $r_{min} = 20$ nm. Simulation **a** and **b** have a persistence length of $L_P = 3.34$ μm, while **c** has a persistence length of $L_P = 33.4$ μm. Filaments in **a** are 200 nm long, while **b** and **c** are 1 μm long.

METHODS

Radial-pair distribution function

The Radial-pair Distribution Function (RDF) is a staple analytical technique [13] in Molecular Dynamics that describes the mass-density at various length scales as a radially symmetric probability density function. Applied to a simulated network, this is useful for identifying filament packing structure (i.e. filament arrangement within a bundle), bundle thickness, and network mesh size.

The RDF, *G(r)*, is calculated as a summation over every bead *k* for *N* beads in volume *V*, with bin size *dr* and *n(r)* beads in the spherical shell of width *dr* and radius *r* centered at *k*:

$$G(r) = \frac{1}{N}\sum_{k=1}^{N} g_k(r) \tag{1}$$

$$g_k(r) = \frac{n(r)}{4\pi r^2 dr}\left(\frac{N}{V}\right) \tag{2}$$

A few additional cautions are important when applying this to a simulated network. The bead-bead spacing along a filament is a highly ordered structure that would leave an artifact in the RDF. To eliminate this, spline interpolation can be used to the extent that the interpolated bead-bead spacing is equal the spatial resolution of the RDF. In the calculation of g(r) for some bead k, beads on the same filament as k can be excluded from the g(r) to remove the signature of a filament's average contour from the final RDF. This provides a clearer representation of the

network structure. The spatial periodicity of the system must also be accounted for when determining the distance between beads. The nearest of the nearest-neighbor images must be used.

Computational cost for these calculations scales with N^2, and spline interpolation effectively increases *N* significantly, further increasing cost. To improve efficiency, GPU parallel processing is suggested to decrease computation time.

In Figure 2 a set of three example RDFs are shown representing different structures. The 2D hexagonal filament packing structure is easy to identify by the ratios of the peak values: 20, 35, 40, and 53 nm (20, $20\sqrt{3}$, $20 \cdot 2$, $20\sqrt{7}$, $20 \cdot 4$). The magenta RDF drops below 1 at 190 nm, representing a void space in which there is a lack of mass at that range (see diagram of Figure 2). In contrast, the cyan and blue RDFs approach 1 smoothly, representing a connected network of filaments. The relative weakness and lack of hexagonal peaks in the blue RDF indicates less filaments per bundle.

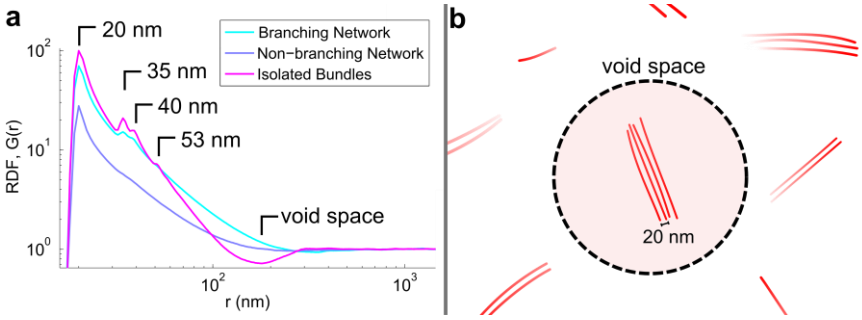


Figure 2. Selected RDFs (**a**) calculated for the three simulated network structures in Figure 1 with logarithmic axes. The adjacent diagram (**b**) demonstrates the void space where the magenta RDF drops below 1; this represents a lack of mass at this range characteristic of isolated bundles of filaments. Peak values are marked for the magenta and cyan RDFs.

Scaling analysis

The fractal dimension (or 'mass-density power-law scaling') of a structure represents the way mass scales with space. Common simple geometries have fractal dimension values which serve as a good reference when looking at more complex structures. For example, a straight line (i.e. a single filament) in 3D has a fractal dimension of $D = 1$. A plane, $D = 2$ (i.e. a thin sheet of material) and a volume, $D = 3$ (i.e. a uniform solid) have similarly simple values. More complex structures often have intermediate values that can be related to these simple geometries. The Koch Curve [14], commonly depicted as a line in 2D with fractally dense curves, has a fractal dimension slightly more than that of a line in 2D ($D \approx 1.262$), since it fills space more completely. The freely jointed chain, a simple model for flexible polymers, is a random walk in 3D with a fractal dimension value intermediate between a volume and line ($D = 2$).

Finite geometries (non-fractal) can be analyzed in the same way as fractal structures, though the range of consistent scaling is limited. Instead, the average mass-density at different size scales will fit a power-law trend when scaling is consistent. This can be applied to our coarse-grained simulation results and one formalism to quantify this is:

$$D = \frac{d \log L(r)}{d \log r} \tag{3}$$

for average mass L at radius r from any bead. This is closely related to the RDF $G(r)$, as the average mass in a space of radius r around a bead is:

$$L(r) = \frac{V}{N} \sum_{\rho=0}^{r} G(\rho) \tag{4}$$

When this technique is applied to a filament network, in the very short range ($r < r_{min}$) the geometry of individual filaments is dominant ($D \rightarrow 1$) and in very long range the space is filled volumetrically ($D \rightarrow 3$). In the intermediate range, the scaling parameter will transition to an intermediate value that may be consistent over a decade. The locations of transition points representing the limits to the length-scale range of a structure can be identified by finding crossover points (i.e. points where D transitions from one value to another).

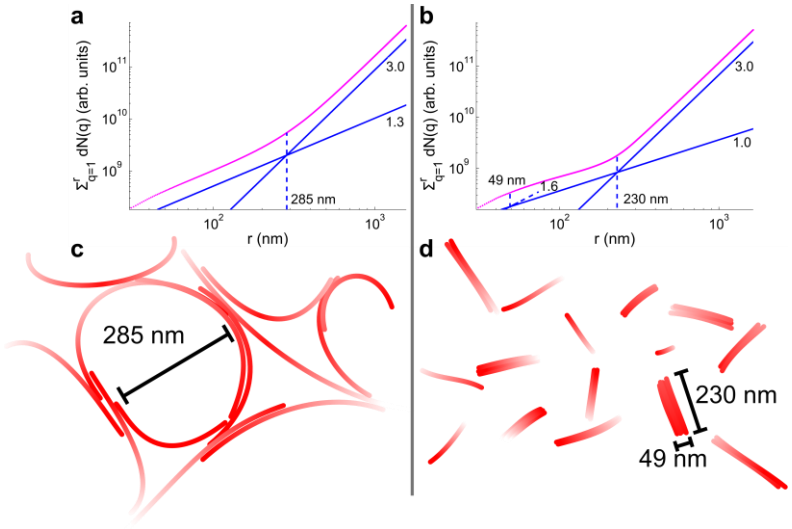


Figure 3. Mass-scaling of a branching network structure (**a**) and isolated bundles (**b**) with different transition points and scaling parameters. Diagrams representing the calculated values in each structure are shown (**c-d**).

In Figure 3**a**, a power-law fit of the mass-scaling for the branching network in Figure 1**b** is shown with a scaling parameter of 1.3 in the intermediate region. The transition point where two power law fits meet represents the mesh size here. The system of isolated bundles in Figure 1**a** similarly has a transition point in its mass-scaling (Figure 3**b**) representing the size of a bundle. Diagrams representing the meaning of these values are shown adjacent in Figure 3 **c-d**.

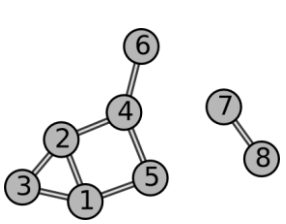
Percolation analysis

An analysis of the connectedness of a simulated network is particularly useful for time-resolved studies of the evolution of the structure, and allows verification that the structure has

evolved to an equilibrium or quasi-equilibrium state. A structure that appears well developed by its RDF may continue to evolve subtly by slowly forming a more connected structure while maintaining its general morphology.

In the evolution of a typical network structure, three distinct time-regimes are identifiable: rapid network formation, slow bundle thickening, and the equilibrated state. In the initial stage, rapid network formation, filaments quickly form 'clusters' of interconnected (directly or indirectly) filaments. The average number of filaments in a cluster grows as separate clusters join, eventually forming a percolated structure with nearly all filaments connected to every other filament by cluster-cluster aggregation [2,15]. Once the network structure is formed, loose ends of filaments continue to fluctuate, forming new connections as they are attracted towards other bundles. At the limit of this process, a quasi-equilibrium structure is formed which persists. This is seen experimentally with F-actin gels, which form a pseudo-stable gel rapidly [2].

To perform the percolation analysis, two parameters are calculated: the average cluster size and the average connections per filament (connectivity). In the first phase of this calculation, the inter-filament "cross-links" are identified by the proximity of close filaments. If any two beads on different filaments are within slightly more than the bond distance ($r_{min} + 1\ nm$), a bond is recorded in a symmetric logic table *T*. If filament number *A* is sufficiently proximate (cross-linked) to *B*, then $T_{A,B}$ is set to 1, otherwise it is 0. The logic table formed by repeating this check for every filament pair is an $N \times N$ symmetric adjacency matrix with all cells along the diagonal set to 1. The upper triangular part is used to prevent redundancy. By taking advantage of the MDAnalysis Python toolbox, this table can be formed efficiently.



$$\begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ & & 1 & 0 & 0 & 0 & 0 & 0 \\ & & & 1 & 1 & 0 & 0 & 0 \\ & & & & 1 & 1 & 0 & 0 \\ & & & & & 1 & 0 & 0 \\ & & & & & & 1 & 1 \\ & & & & & & & 1 \end{bmatrix}$$

Figure 4. An example network graph and corresponding adjacency matrix. The network has an average connectivity value of 0.609 and average cluster size of 0.625.

As an example of this process, consider the example network graph in Figure 4 representing eight filaments with "cross-links" between them. The logic table representing this network graph is shown adjacent. Each diagonal cell uniquely represents a node/filament, while off-diagonal cells represent connections between nodes/filaments.

In the second phase of calculation, a recursive algorithm parses this logic table incrementing a counter for each value. The recursive algorithm is simply stated for any cell $T_{i,j}$:

- If $i = j$, increment the cluster size counter
- If $i \neq j$, increment the connectivity counter
- Set $T_{i,j} = 0$
- For each cell $T_{i',j'} = 1$ such that $i' = i$ or $j' = j$ (on the same row or column), recurse from that cell.

The recursion is initialized from each cell on the diagonal, each time with the original copy of *T*. After recursion collapses, the cluster size and connectivity counters are normalized by the maximum value of cluster size, N^2. This recursion thus follows the connections between nodes, going deeper into the network graph with each depth of recursion until a full cluster is parsed. Since each diagonal cell parsed represents a filament connected (either directly or indirectly) to the cluster which includes the initialized node, by counting these diagonal cells the size of the cluster can be determined. Similarly, counting off-diagonal cells gives a measure of how well connected the cluster is. A more dense adjacency matrix will then have a higher connectivity value, while a more sparse adjacency matrix will have a lower connectivity value. A minimally connected full network (ring topology) would have a normalized connectivity value of 1, thus values greater than 1 are easily possible.

An example of this recursive algorithm is represented in Figure 5 for the network shown in Figure 4. Recursion is initialized at cell $T_{5,5}$ (representing node 5) as an example. At the top depth of recursion (red), $T_{5,5}$ is set to 0, and the cluster size counter is incremented once. The four nonzero cells in the row/column begin the next depth of the recursion (shown as dark blue). Cells in the same rows/columns are again parsed (light blue) looking nonzero values for the next depth of recursion etc. At this depth, the cluster size counter is incremented twice, while the connectivity counter is incremented four times. In this case, recursion reaches a depth of 4. The remaining nonzero values represent nodes 7 and 8, which are unconnected to the larger cluster; they contribute nothing in this case.

The computational cost of this procedure scales poorly, with N^3. This can be improved by under-sampling the number of diagonal cells from which it is initialized.

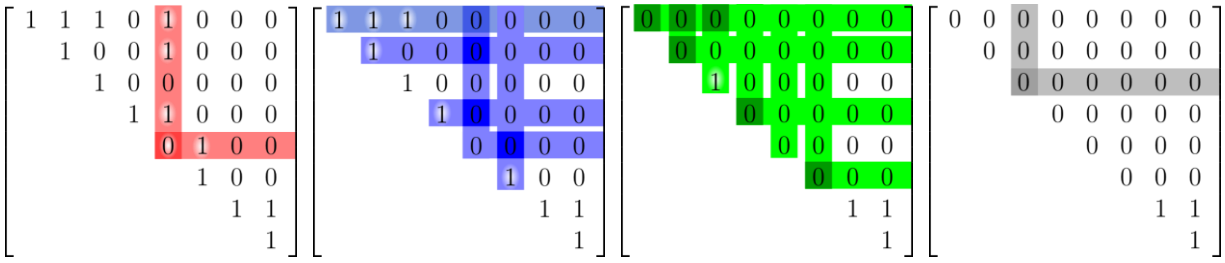


Figure 5. Example steps in computing connectivity and cluster size for the network in Figure 4 initialized from node 5. Recursion depth is represented as different colors (increasing from left to right). Dark colored cells represent points of recursion, while light colors represent cells parsed for the next depth of recursion. Parsed 1's continue the recursion procedure. Same-depth recursion is shown in parallel here for conciseness, while a series programming approach is necessary to prevent double-counting cells.

CONCLUSIONS

We have presented a toolbox of techniques for the analysis of simulated semi-flexible polymer networks. The RDF, Scaling analysis, and Percolation analysis allow quantitative characterization of important structural properties. It is expected that interest in applications of tunable synthetic filaments will require a computational model and metrics for comparison. This set of techniques allows quantitative measurement of values useful in defining network structure.

These methods serve as a basic toolbox for reporting structural properties in future filament network simulations.

ACKNOWLEDGMENTS

We acknowledge L. T. Nguyen for his contribution of the coarse grain F-actin model and simulation framework. We acknowledge financial support from NSF CAREER grant #DMR-0852791 and grant #DMR-0843934.

REFERENCES

1. H. Lodish, A. Berk, S. L. Zipursky, P. Matsudaira, D. Baltimore, and J. Darnell., *Molecular Cell Biology*, 4th ed. (Freeman, New York, 1999).
2. L. S. Hirst and C. R. Safinya, *Phys. Rev. Lett*. **93**, 018101 (2004).
3. O. Pelletier, E. Pokidysheva, L. S. Hirst, N. Bouxsein, Y. Li, and C. R. Safinya, *Phys. Rev. Lett*. **91**, 3 (2003).
4. P. H. J. Kouwer, M. Koepf, V. a a Le Sage, M. Jaspers, A. M. van Buul, Z. H. Eksteen-Akeroyd, T. Woltinge, E. Schwartz, H. J. Kitto, R. Hoogenboom, S. J. Picken, R. J. M. Nolte, E. Mendes, and A. E. Rowan, *Nature* **493**, 651 (2013).
5. N. A. Peppas, J. Z. Hilt, A. Khademhosseini, and R. Langer, *Adv. Mater*. 18, 1345 (2006).
6. A. R. Hirst, B. Escuder, J. F. Miravet, and D. K. Smith, *Angew. Chem. Int. Ed. Engl*. **47**, 8002 (2008).
7. J. C. Tiller, *Angew. Chem. Int. Ed*. Engl. **42**, 3072 (2003).
8. E. S. Place, N. D. Evans, and M. M. Stevens, *Nat. Mater.* **8**, 457 (2009).
9. A. Agrawal, N. Rahbar, and P. D. Calvert, *Acta Biomater*. **9**, 5313 (2013).
10. L. T. Nguyen, W. Yang, Q. Wang, and L. S. Hirst, *Soft Matter* **5**, 2033 (2009).
11. L. T. Nguyen and L. S. Hirst, *Phys. Rev. E* **83**, 1 (2011).
12. R. J. Pandolfi, L. Edwards, D. Johnston, P. Becich, and L. S. Hirst, in press *Phys. Rev. E* (2014).
13. M. De Podesta, *Understanding the Properties of Matter* (CRC Press, 2002).
14. B. B. Mandelbrot, *The Fractal Geometry of Nature* (Macmillan, 1983).
15. L. S. Hirst, R. Pynn, R. F. Bruinsma, and C. R. Safinya, *J. Chem. Phys.* **123**, 104902 (2005).