To implement the cluster-cluster aggregation algorithm, we need to store the position of each particle and the cluster to which each particle belongs. In class `CCA`, which can be downloaded from ch13 directory, the position of a particle is given by its x- and y-coordinates and stored in the arrays `x` and `y`, respectively. The array element `site[x][y]` equals zero if there is no particle at (x, y); otherwise, the element equals the label of the cluster to which the particle at (x, y) belongs.

The labels of the clusters are found as follows. The array element `firstParticle(k)` gives the particle label of the first particle in cluster `k`. To determine all the particles in a given cluster, we use a data structure called a *linked list*. We implement the linked list using the array `nextParticle`, so that the value of an element of this array is the index for the next element in the linked list. The array `nextParticle` contains a series of linked lists, one for each cluster, such that `nextParticle[i]` equals the particle label of another particle in the same cluster as particle `i`. If `nextParticle[i]` $= -1$, there are no more particles in the cluster. To see how these arrays work, consider three particles 5, 9, and 16 which constitute cluster 4. We have `firstParticle[4]` $= 5$, `nextParticle[5]` $= 9$, `nextParticle[9]` $= 16$, and `nextParticle[16]` $=$ `-1`.

As the clusters undergo a random walk, we need to check if any pair of particles in different clusters have become nearest neighbors. If such a situation occurs, their respective clusters have to be merged. The check for nearest neighbors is done in method `checkNeighbors`. If `site[x][y]` and `site[x+1][y]` are both nonzero and are not equal, then the two clusters associated with these sites need to be combined. To do so, we add the particles of the smaller cluster to those of the larger cluster. We use another array, `lastParticle`, to keep track of the last particle in a cluster. The merger can be accomplished by the following statements:

```
// link last particle of larger cluster to first particle of smaller
// cluster
nextParticle[lastparticle[largerClusterLabel]] =
     firstParticle[smallerClusterLabel];
// sets the last particle of larger cluster to the last particle of
// smaller cluster
lastParticle[largerClusterLabel] = lastParticle[smallerClusterLabel];
// adds mass of smaller cluster to the larger cluster
mass[largerClusterLabel] += mass[smallerClusterLabel];
```

To complete the merger, all the entries in `site[x][y]` corresponding to the smaller cluster are relabeled with the label for the larger cluster, and the last cluster in the list is relabeled by the label of the small cluster, so that if there are n clusters they are labeled by $0, 1, \ldots, n-1$.

(a) Write a target class for class `CCA`. The class assumes that the diffusion coefficient is independent of the cluster mass. Choose $L = 50$ and $N = 500$ and describe the qualitative appearance of the clusters as they form. Do they appear to be fractals? Compare their appearance to DLA clusters.

(b) Compute the fractal dimension of the final cluster. Use the center of mass $\mathbf{r}_{\text{cm}}$ as the origin of the cluster, where $\mathbf{r}_{\text{cm}} = (1/N)\left(\sum_i x_i, \sum_i y_i\right)$, and (x_i, y_i) is the position of the ith particle. Average your results over at least ten final clusters. Do the same for other values of L and N. Are the clusters formed by cluster-cluster aggregation more or less space filling than DLA clusters?

(c) Assume that the diffusion coefficient of a cluster of s particles varies as $D_s \propto s^{-1/2}$ in two dimensions. Let D_{max} be the diffusion coefficient of the largest cluster. Choose a random number r between 0 and 1 and move the cluster if $r < D_s/D_{\text{max}}$. Repeat the simulations in part (a) and discuss any changes in your results. What effect does the dependence of D on s have on the motion of the clusters? ■

REFERENCES AND SUGGESTIONS FOR FURTHER READING

We have considered only a few of the models that lead to self-similar patterns. Use your imagination to design your own model of real-world growth processes. We encourage you to read the research literature and the many books on fractals.

R. C. Ball and R. M. Brady, "Large scale lattice effect in diffusion limited aggregation," J. Phys. A **18**, L809–L813 (1985). The authors discuss the optimization algorithm used in Project 13.17.

Albert–László Barabási and H. Eugene Stanley, *Fractal Concepts in Surface Growth* (Cambridge University Press, 1995).

J. B. Bassingthwaighte, L. S. Liebovitch, and B. J. West, *Fractal Physiology* (Oxford University Press, 1994).

D. Ben-Avraham and S. Havlin, *Diffusion and Reactions in Fractals and Disordered Systems* (Cambridge University Press, 2005).

K. S. Birdi, *Fractals in Chemistry, Geochemistry, and Biophysics* (Plenum Press, 1993).

Armin Bunde and Shlomo Havlin, eds. *Fractals and Disordered Systems*, revised edition (Springer–Verlag, 1996).

Fereydoon Family and David P. Landau, eds., *Kinetics of Aggregation and Gelation* (North–Holland, 1984). A collection of research papers that give a wealth of information, pictures, and references on a variety of growth models.

Fereydoon Family, Daniel E. Platt, and Tamás Vicsek, "Deterministic growth model of pattern formation in dendritic solidification," J. Phys. A **20**, L1177–L1183 (1987). The authors discuss the nature of Laplace fractal carpets.

Fereydoon Family and Tamás Vicsek, eds., *Dynamics of Fractal Surfaces* (World Scientific, 1991). A collection of reprints.

Fereydoon Family, Y. C. Zhang, and Tamás Vicsek, "Invasion percolation in an external field: Dielectric breakdown in random media," J. Phys. A. **19**, L733–L737 (1986).

Jens Feder, *Fractals* (Plenum Press, 1988). This text discusses the applications as well as the mathematics of fractals.

Gary William Flake, *The Computational Beauty of Nature* (MIT Press, 2000).

J.-M. Garcia–Ruiz, E. Louis, P. Meakin, and L. M. Sander, eds., *Growth Patterns in Physical Sciences and Biology* (NATO ASI Series B304, Plenum, 1993).

Peter Grassberger, "Critical percolation in high dimensions," Phys. Rev. E **67**, 036101-1–4 (2003). The author uses the Leath algorithm to estimate the value of p_c.

Thomas C. Halsey, "Diffusion limited aggregation: A model for pattern formation," Physics Today **53** (11), 36 (2000).