COMPUTER SIMULATIONS

Department Editors:
Harvey Gould
Jan Tobochnik

Simulation of Diffusion-Controlled Chemical Reactions

Panos Argyrakis

Chemical reactions in condensed phases have attracted considerable interest,[1–4] since it was discovered[3,4] that the known laws of kinetics apply only to the homogeneous three-dimensional space of a reaction vessel or test tube. As disordered spaces and spaces of low dimensions became more fashionable and more manageable, several new approaches and problems have evolved, and we will see that the previous understanding of diffusion-controlled chemical reactions has to be modified.

Let us consider an example of a prototype reaction. The usual textbook kinetic rate law for the second-order reaction $A + A = 0$ is expressed as[4]

$$\frac{dA}{dt} = -kA^2 \qquad \text{(batch mode)} \tag{1a}$$

$$\frac{dA}{dt} = 0 = r - kA^2 \qquad \text{(steady-state mode),} \tag{1b}$$

where A is the density of A particles at time t, k is the rate constant, and r is the creation rate of the A particles. Such a reaction can occur in either "batch" or "steady-state" mode. In the batch (or transient) mode, all reactants are entered into the system at time $t = 0$, usually at random positions, and no reactants enter at $t > 0$. In this mode the density of A decreases with t. This mode is relevant when the main quantity of interest is the reaction rate constant k. In the steady-state mode, the same reaction mechanism holds, but new reactants are entered into the system during the course of the reaction at regular time intervals. In this mode there is no density decay with increasing t, and the main interest is the way that the system does or does not attain an equilibrium state. The differential equation (1) has the simple solution

$$1/A - 1/A_0 = kt \quad \text{(batch mode)}, \tag{2a}$$

$$A = \sqrt{r/k} \quad \text{(steady-state mode)}, \tag{2b}$$

where A_0 is the density at time $t = 0$.

For heterogeneous spaces and dimensions less than three, the rate equation (1) and hence the solution (2) must be modified. To understand these modifications, we consider some of the properties of a single random walker. In uniform systems, the walker's root-mean-square displacement, $R \equiv \sqrt{\langle R^2(t) \rangle}$, is proportional to $t^{1/2}$ for any integer spatial dimension d. However, on a fractal lattice such as a percolation cluster, this relation is not applicable, and the relation becomes, in general, $R \sim t^{1/d_w}$ for sufficiently large t. The exponent d_w is the fractal dimension of the path of the random walker on the fractal lattice. In general, $d_w > 2$, because the random walker is slowed down by the dangling ends and bottlenecks in the disordered lattice. Another quantity of interest is $S(t)$, the number of distinct sites that have been visited at time t by the random walker. After t steps, $S(t)$ is proportional to the volume F^{d_f} $\sim t^{d_f/d_w}$. The ratio d_f/d_w can be shown to be related to the density of vibrational excitations on the fractal lattice, that is, $\rho(\epsilon) \sim \epsilon^{d_f/d_w - 1} = \epsilon^{d_s/2 - 1}$, where the "spectral" dimension d_s is given by $d_s = 2d_f/d_w$. The dependence of $\rho(\epsilon)$ on ϵ is similar to the expression for the density of states in Euclidean space, $\rho(\epsilon) \sim e^{d/2 - 1}$, with d replaced by d_s.

Because the diffusion of a random walker on a fractal lattice is modified, we expect that the relation (1) also is modified. We find that

$$\frac{dA}{dt} = -k_0 t^{-h} A^2 \quad (0 \leqslant h < 1) \quad \text{(batch mode).} \tag{3a}$$

and

$$\frac{dA}{dt} = 0 = r - k_0 A^x, \quad x = \frac{2-h}{1-h} \quad \text{(steady-state mode)}. \tag{3b}$$

Equation (3) is applicable to $d = 1$, $d = 2$, and fractal spaces with noninteger dimensionality. The exponent h in (3) is given by $h = 1 - d_s/2$. Such a relation between h and d_s might be expected, because it is well known that a random walk models diffusion processes, such as the diffusion of reacting particles. The solution of the differential equation (3a) is

$$1/A - 1/A_0 = k_0 t^{d_s/2}. \tag{4}$$

Note the difference in the exponent of t between the solutions (2a) and (4).

Some simple cases can be quickly summarized. For $d = 1$ lattices, $d_s = 1$, and, consequently, $h = 1/2$ and $x = 3$. For percolation clusters in two and three dimensions, $d_s \approx 4/3$, and hence $h \approx 1/3$, and $x \approx 2.5$. These results can be verified by computer simulations.[4–6]

Another interesting bimolecular reaction is the $A + B = 0$ reaction. For homogeneous lattices the solution of the rate law (for the case of $A_0 = B_0$) is given by[1]

$$\frac{1}{A} - \frac{1}{A_0} = k_0 t^{d/4}, \tag{5}$$

yielding exponents 0.25, 0.50, and 0.75 for $d = 1$, $d = 2$, and $d = 3$ lattices, respectively. For percolation clusters in

Panos Argyrakis is Associate Professor of Physics, University of Thessaloniki, 54006 Thessaloniki, Greece; E-mail: caczl1@grtheun1.

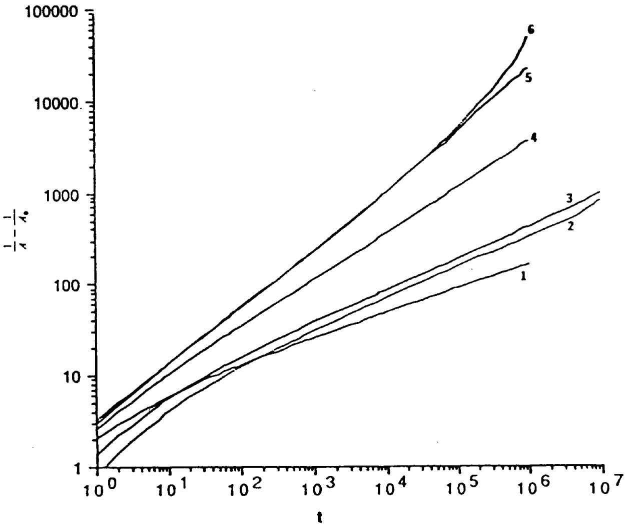

FIG. 1. Plot of $1/A - 1/A_0$ versus the time t. Curves 1–3 are for the $A + B$ reaction with (1) $d = 1$ ideal lattice, (2) $d = 2$ percolation cluster, and (3) $d = 3$ percolation cluster. Curves 4–6 are for the $A + A$ reaction with (4) $d = 1$ ideal lattice, (5) $d = 2$ percolation cluster, and (6) $d = 3$ percolation cluster. The lattice sizes were 10^5 sites ($A + B$ for $d = 1$), 2×10^6 sites ($A + A$ for $d = 1$), 2000×2000 for $d = 2$, and $160 \times 160 \times 160$ for $d = 3$. The percolation clusters were generated at the percolation thresholds $p_c = 0.5931$ ($d = 2$), and $p_c = 0.3117$ ($d = 3$). The initial densities were $A_0 = 0.8$ ($A + A$), and $A_0 = B_0 = 0.4$ ($A + B$). A total of 5–25 realizations were used for the average values.

$d = 2$ and $d = 3$, the exponent $d/4$ in (5) is replaced by the exponent[7,8] $d_s/4$, which has a value of approximately 1/3. These exponents can be verified by computer simulation,[8] although with more difficulty than the $A + A$ case. Some typical results[8] for the $A + A = 0$ and the $A + B = 0$ reactions are shown in Fig. 1 for $d = 1$ lattices and for $d = 2$ and $d = 3$ percolation clusters.

An interesting outcome of reactions on fractal lattices is that the rate constant k, which is normally independent of both concentration and time [cf. (1)], loses this property and becomes time dependent. This behavior of the rate constant is related to the rate of exploration of the space by the reacting particles; that is,

$$k \sim \frac{dS(t)}{dt} \qquad (6)$$

This rate is constant only for $d = 3$ (see Suggestions for Further Study).

I now discuss some of the computational techniques that have been developed by our group during the past few years to simulate the reaction mechanism discussed in the above. The problems we need to consider are the fo.lowing.

(1) What data should be kept in memory during the run? Of course, we try to reduce these data to a minimum.

(2) How is time calculated?

(3) How is the problem of multiple occupancy treated? In a reaction two particles cannot normally occupy the same site because they have a finite size. But what happens if this restriction is relaxed so that we allow two or more particles to occupy the same lattice site at the same time?

(4) How is the nearest-neighbor distance between particles calculated for regular lattices (straightforward), and for random lattices, where the direct point-to-point connection is not valid (more difficult)?

It is straightforward to simulate chemical reactions based on diffusion on a lattice.[5,8,9] A population of reacting particles is monitored is some region of space as a function of time. The positions of the particles are known. Each particle performs a random walk independently of all other particles. Steps are normally allowed to nearest-neighbor sites only. There is no interparticle interaction. A reaction occurs when two particles collide, i.e., land at the same site. Every collision leads to a reaction with probability one. When two particles react in this fashion, they are removed from the system (they are annihilated).

A reaction can be first-, second-, or higher-order depending on the power dependence of the reactant concentrations. If only one kind of reactant is present, we call this reaction the $A + A$ type, because it takes two A particles (two similar particles) for a reaction to occur. Thus the reaction is second order in A and second order overall. The $A + A$ reaction can lead to either $A + A = 0$ or to $A + A = A$; that is, both or only one of the two A particles are annihilated. If there are two types of reacting particles present, then the reaction is the $A + B$ type. In this case two A's or two B's do not react, but A reacts with B. The $A + B$ reaction is also second order overall (it takes two particles for a reaction to occur), but is first order in A and first order in B.

The lattice can have any dimension, e.g., a line, a surface, a fractal, or any other well-defined structure. The main observation of interest is how the rate laws depend on the lattice structure, giving, for example, different exponents, or some new characteristic behavior, as, for example, the fact that the segregation of the A and B particles in the $A + B$ reaction occurs in $d = 2$ but not in $d = 3$ (see problem 6 in Suggestions for Further Study).

First, we consider the problem of how to treat the reactants. It is clear that the two reaction types, $A + A$ and $A + B$, must be treated differently. The problem of the excluded volume only appears in the latter case, and its treatment also is different when any number of particles may occupy a given site. Consider the case of the $A + A$ reaction on a lattice. At time $t = 0$ (in batch mode), all the reactants are placed on the lattice, e.g., their coordinates are decided by use of a random-number generator. The same considerations apply if the reaction is of the steady-state type, for which the reactants also enter the system at random positions at given time intervals. One array for each dimension keeps the coordinates of all the particles. For example, for a two-dimensional reaction, $NX(5) = 17$ and $NY(5) = 3$ means that the coordinates of particle 5 are stored in two arrays, NX and NY, and the coordinates of particle 5 are (17,3). The lattice is kept in memory in a separate array. Each lattice site has an integer value that signifies the number of the particle. Thus $M(17,3) = 5$ shows that site (17,3) is occupied by particle number 5. If $M(i,j) = 0$, the site (i,j) is not occupied by any particle.

The simulation proceeds by running through the particle list and moving one particle at a time in a random direction from (i,j) to (i',j'). If $M(i',j') \neq 0$, then the new site (i',j') is already occupied and both particles must be annihilated. This procedure is done by setting both $M(i,j)$ and $M(i',j')$ equal to zero. Also, the two arrays NX and NY must be changed. The easiest way to make this change is to give the last two particles in the list the labels of the two annihilated particles. Thus, if any time we have 100 particles and we annihilate particle 8 and particle 30, we would set $NX(8) = NX(100)$, $NY(8) = NY(100)$, $NY(30) = NY(99)$, and $NY(30) = NY(99)$.

It might seem redundant to keep the same information in two separate arrays. A simple alternative is to keep only the array M and to refer to this array when we need to know the occupancy of a given site. However, this latter method is not the fastest if we consider the nature of the reaction process. In the suggested method the list of all remaining particles at a given time is run through once, because it is straightforward after moving a particle to determine its new location. If this list were not kept in memory, then we would have to locate and move the reacting particles by sweeping through the entire lattice at each time step. This sweep is considerably more time consuming, especially when only a few particles are present. In other words, going through the list of particles rather than going through all lattice sites is more efficient.

For the $A + B$ reaction, we must modify our procedure slightly. We keep separate lists for the A and B particles, with the value of the array again denoting the label of the particle. The $M(i,j)$ array is used as before, but the array values can now be positive or negative, e.g., if the array is negative the particle is type A, and if it is positive it is type B. For example, if $M(17,3) = -5$, the site with coordinates (17,3) is occupied by the A particle with particle label five. Because of the excluded-volume principle, a particle of one type is not allowed to occupy a site occupied by another particle of the same type. We can implement the exclusion principle in two ways. One way is to allow the random walker to choose a new site freely, but to return the walker to its previous location if it chooses a site that is already occupied by a particle of the same type. This type of random walker is called a "blind ant." The alternative is to allow the walker to choose only between sites that are unoccupied or have a particle of the other type. This latter type of walker is called a "myopic ant." We also use the blind-ant condition for the random lattices when a walker attempts to move off the percolation cluster.

The calculation of time. There are two ways of monitoring the time during the course of a reaction. The first way is to consider that one reaction step is completed when all remaining particles have moved one nearest-neighbor distance. Here, the time clock is independent of the number of annihilations occurring in one sweep through the particle list. In the second method the clock is increased by $1/N$ time units, where N is the number of particles present after each particle makes a move. Here, the time increment is a function of the number of particles present. For bookkeeping purposes the first method is easier to implement. In the second method, time is a real number, and when we want to calculate averages or find the particle density at fixed intervals of time, we have to bin the times at the desired intervals. The methods yield noticeable deviations (for example, in the density decay) only at early times, but at long times they give practically identical results.

The multiple occupancy problem. When more than one particle is allowed to occupy the same site, we have to follow another strategy. We employ the following method and still avoid sweeping through the lattice. Initially, all sites are zeroed; i.e., the $M(i,j)$ are assigned a value of 0. The appropriate number of particles are then placed on the lattice, with more than one particle allowed on a site. Every site has a value that is the algebraic sum of all particles on that site; that is, each A particle contributes -1 and each B particle contributes $+1$. Thus, if there happen to be three A particles and one B particle on site (i,j), then $M(i,j) = -2$. The annihilation of reacting particles throughout the lattice is done as follows. First, we check for the A particles by going through the particle list (which contains the particle coordinates) and checking the value of the array M at the site where a given A particle is located. This procedure is done in the same way as before; i.e., one particle is moved at a time, and a check is performed after each particle move. If the value is zero or positive, the A particle is annihilated, because there must be at least one B

particle at this site. If the value is negative, the A particle stays intact, and the value is increased by one. After we finish going through the list of A particles, we go through the B particle list in the same manner. Here, the B particle is annihilated if the value of M is zero or negative. If it is positive, the B particle stays intact, and M at that site is decreased by unity. The increase or decrease by unity when no particles are annihilated automatically zeroes all the values of M after going through the particle lists. After the reaction step is completed, the particles are allowed to diffuse one step and the array M is filled by adding -1 to $M(i,j)$ each time an A particle lands on site (i,j) and by adding $+1$ each time a B particle lands on site (i,j).

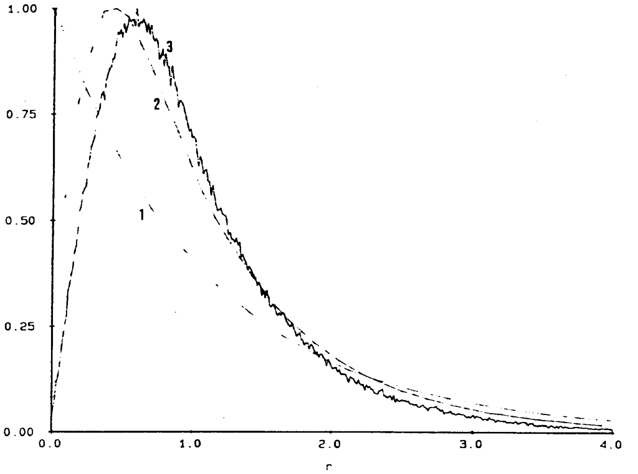

FIG. 2. The nearest-neighbor distance distributions for the $A+B$ reaction in $d=1$. All distances r are the AB nearest neighbor distances. The horizontal axis is the normalized distance $(r-1)/(\langle r\rangle - 1)$, where $\langle r\rangle$ is the average distance. The Poissonian curve (line 1) is for $t=0$, and the Wigner-type curves are for $t=10$ steps (line 2) and $t=1000$ steps (line 3). The lattice dimension is $L=10^4$ sites, and $A_0=B_0=0.1$. All data shown are averages of 1000 realizations.

To see how the reaction process works, consider the example of three A particles and one B particle at a site (i,j) so that $M(i,j) = -2$. When we find the first A particle at site (i,j), $M(i,j)$ will be increased to -1. When we find the second A particle, $M(i,j)$ will be increased to 0. When we find the third A particle, it will be eliminated because $M(i,j)=0$. When we find the B particle at (i,j), it will also be eliminated. Thus, at the end of this reaction step, the correct numbers of A and B particles have been eliminated and $M(i,j)=0$.

The problem of finding the nearest-neighbor distances. Our understanding of the mechanism of a reaction is enhanced if we know the distribution of the nearest-neighbor distances of all reacting particles. The nearest-neighbor distance of a particle is defined as the smallest distance between the particle and all other particles. The distribution of these distances changes dramatically as the reaction proceeds, and this change can provide information about the reaction mechanism. At time $t=0$, the most probable nearest-neighbor distance between randomly placed particles in any dimension is $r=1$ (one lattice constant). The distribution of nearest-neighbor distances on a $d=1$ lattice is given by Hertz[10]

$$P(r) = 2\rho e^{-2\rho(r-1)}, \tag{7}$$

where ρ is the particle density. The distribution $P(r)$ in (7) is shown in Fig. 2. As the reaction proceeds, the distribution of nearest-neighbor distances changes, because the $r=1$ neighboring particles have a higher probability of reacting, because the remaining particles now are further apart, the average distance and the most probable distance increase. The distribution at a later time also is shown in Fig. 2. We see a crossover from an exponentially decreasing curve at $t=0$ to a Gaussian-like curve with a maximum. Analogous formulas and distributions are applicable to $d=2$ and $d=3$ lattices.

The observed crossover is a key characteristic of the extent of the reaction, and therefore the computation of the $P(r,t)$ distribution is important. By definition, each remaining particle has only one nearest-neighbor distance, and the problem is to find the distribution of all such distances. For a perfect lattice this calculation is not difficult. We simply place the origin at a given particle, and then move away from the particle, one lattice spacing at a time, until the first new particle is found, and record this distance.

For random lattices, such as a percolation cluster, this method does not work, as we can see from Fig. 3. The shaded area is not part of the cluster, and thus the distance between A and B is not a straight line, but a line that goes around the shaded area. The algorithm to find the length of this line is more complicated and is called breadth-first search.[11] In this method we look for the nearest-neighbor of a given particle by first looking at the immediate neighboring sites, ignoring sites that do not belong to the percolation cluster. The percolation clusters are constructed using the well-known cluster-multiple-labeling technique,[12] which will not be described here. These first neighbor sites are labeled as visited by this search, and their (x,y) coor-

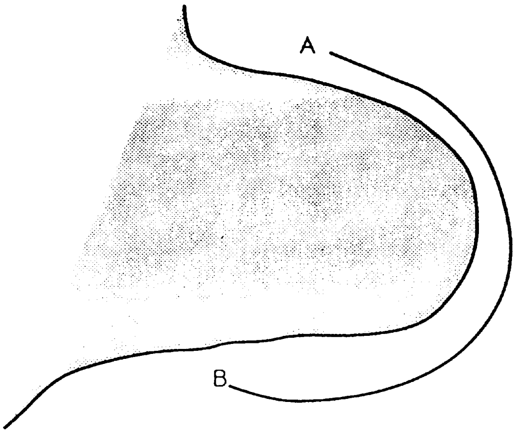

FIG. 3. Schematic of a section of a percolation cluster. We want to calculate the nearest distance between points A and B on the cluster. Because the shaded area is not part of the cluster, the answer is not the straight line connecting points A and B, but the curved line shown that goes around the shaded area.

dintes and their r distance from the particle of interest are placed in an array. If none of these sites contains a reacting particle, then we next look for the nearest-neighbors of these first neighbors that have not yet been visited. Their (x,y) coordinates and their distances r are then placed at the end of the array, one site at a time. If none of these second neighbors contains a reacting particle, we continue on the third neighbors and so on. Every new site that is considered is labeled as visited. Its distance r is simply the distance of the previous (mother) site plus one. In this way the length of the array grows quite fast, but it is still the most efficient method for solving this problem. In this method a traveling wave (breadth) is formed in all directions away from the original point, but it covers only the region of the cluster. It follows all the random paths of the percolation cluster, its bottlenecks, and all types of clusters regardless of any peculiarities that might occur. This process is continued until the first reacting particle is encountered. This particle is by definition the nearest neighbor. Its distance is recorded and used to obtain $P(r)$. It is obvious that the distance between two particles whose direct path is blocked by a site that is not part of the cluster is calculated around this site and not over it.

The above techniques are just a fewof the ones used to simulate chemical reactions. The literature I have cited contains several alternate approaches (see Ref. 5 for example). There is a wealth of current problems that are being investigated by several groups interested in modeling chemical reactions. Their goal is to develop more-realistic models and to make a direct comparison with experimental data. For example, frequently the different reactants do not have the same mobility, as the simple model discussed in the text assumes. Because an encounter between particles does not always lead to a reaction, it would be more realistic to assume a reaction probability of less than unity, depending on the conditions. A more-realistic model also might assume the presence of an interparticle interaction and that reactions would occur depending on the distance between reactants and on the strength of the interaction. More-sophisticated models also could include a bias in the particle motion, such as the incorporation of an electrostatic charge on each particle. The field of chemical reactions is rich, and the possibilities of future investigations are practically endless.

Suggestions for further study

(1) Use the techiques discussed in the text to write a program to simulate the batch reaction $A + A = 0$ on a $d = 1$ lattice. Use a line of length $L = 10^4$ sites and cyclic boundary conditions; i.e., set $L(10001) = L(1)$. Begin with $A_0 = 1.0$; that is, fill all lattice sites with A particles. Plot the quantity $(1/A - 1/A_0)$ on log–log paper as a function of time (number of steps) and calculate the slope of the resulting straight line. Compare the slope with the number of visited sites of exponent. [Recall that $S(t) \sim t^{1/2}$ for $d = 1$.] How well do the slopes compare?

(2) Do the same problem on a $d = 2$ lattice. Compare your results to the behavior of $S(t)$, where $S(t) \sim t/\log t$ for $d = 2$. How well do the slopes compare? Can you reconcile the difference? Next, do the same problem with $A_0 = 0.01$. What slope do you obtain? What can you conclude about the initial density dependence?

(3) Do the same problem on a $d = 3$ lattice. What is the value of the exponent? How does it compare with the $S(t)$ exponent, where $S(t) \sim t$ in $d = 3$.

(4) Investigate the time dependence of the "rate constant" k for the above three cases. Which one of the three cases yields a real constant?

(5) Verify the result (7) for the nearest-neighbor particle distribution on a $d = 1$ lattice. Then begin with $A_0 = 0.1$ and modify your program for problem (1) to compute the nearest-neighbor distribution after 10, 100, and 1000 steps. What shapes do the four curves have? Are your results in qualitative agreement with Fig. 2?

(6) Prepare a pictorial of the $A + B$ reaction in $d = 2$ to observe the well-known segregation effect in batch mode. Use a graphic screen or a hard copy plotter to follow the reaction. Prepare a lattice of size 100×100 sites, and use empty circles for A and filled circles for B. (If you have a color monitor, use different colors for the A and B particles). Observe the system at $t = 0$, i.e., before the reaction begins. You should see a completely random positioning of the two species. Now let your program perform the reaction, and show the particles after 100, 1000, and 10 000 steps. You should begin to see how clusters of open and full circles are formed. After segregation occurs, which particles are now able to react?

Acknowledgments

Several of the problems discussed here have been investigated in collaboration with R. Kopelman, whom I would like to thank for very useful discussions. The support of NATO Grant CRG 920029 is gratefully acknowledged. I also would like to thank H. Gould and J. Tobochnik for their helpful suggestions. ■

References

1. A. Blumen, J. Klafter, and G. Zumofen, "Models for reaction dynamics in glasses," in *Optical Spectroscopy of Glasses*, edited by I. Zschokke (Reidel, Dordrecht, the Netherlands, 1986), p. 199.
2. G. H. Weiss, J. Stat. Phys. **42**, 3 (1986).
3. D. Toussaint and F. Wilczek, J. Chem. Phys. **78**, 6242 (1983).
4. R. Kopelman, J. Stat. Phys. **42**, 185 (1986).
5. D. Ben-Avraham, J. Chem. Phys. **88**, 941 (1988).
6. D. Ben-Avraham and S. Havlin, Adv. Phys. **36**, 695 (1987).
7. D. Stauffer and A. Aharony, *Introduction to Percolation Theory* (Taylor and Francis, London, 1991), 2nd ed.
8. P. Argyrakis and R. Kopelman, Phys. Rev. A **45**, 5814 (1992).
9. P. Argyrakis and R. Kopelman, Phys. Rev. A **41**, 2114 (1990).
10. P. Hertz, Math. Ann. **67**, 387 (1909); E. W. Montroll and W. W. Badger, *Introduction to Quantitative Aspects of Social Phenomena* (Gordon and Breach, New York, 1974), p. 100.
11. E. Horowitz and S. Sahni, *Fundamentals of Computer Algorithms* (Computer Sciences, Potomac, MD, 1978).
12. J. Hoshen and R. Kopelman, Phys. Rev. A **14**, 3438 (1976).

From the editors. Future columns are planned on world-line quantum Monte Carlo and wavelets. Please address comments and suggestions to hgould @vax.clarku.edu or jant @kzoo.edu.