PHYSICAL REVIEW E VOLUME 49, NUMBER 2 FEBRUARY 1994

Random-adding determination of percolation thresholds in interacting systems

A. Drory and I. Balberg
Racah Institute of Physics, Hebrew University, Jerusalem 91904, Israel

B. Berkowitz
Hydrological Service, Water Commission, Ministry of Agriculture, P.O. Box 6381, Jerusalem 91063, Israel
(Received 21 June 1993; revised manuscript received 18 October 1993)

A generalization of the random-adding procedure for the determination of percolation parameters in interacting systems is demonstrated. This method which simply utilizes the Boltzmann distribution function is shown to reproduce quite accurately results which were obtained previously by the much less efficient and much less direct method of the Metropolis algorithm. The success of the present method is attributed to the fact that, for typical objects considered in such systems, the effect of the square-well attractive interaction compensates for the effect of the (hard-core) repulsive interaction on the spatial distribution of the objects around a given object.

PACS number(s): 64.60.Ak, 61.20.Ja, 82.70.Kj

The theory of percolation [1] made its major advances through lattice models which enabled the predictions of the universal behavior of lattice and nonlattice systems in the vicinity of the percolation threshold. On the other hand, system-dependent properties, such as the percolation thresholds themselves, require models which describe the system more accurately [2–5]. The theory of connectivity and its physical implications in systems composed of objects or particles is known as the theory of continuum percolation [3]. In recent years progress has been made in considering such systems when the percolating phase is made of permeable [6–10] or partially permeable [7,9] objects. In the latter case most of the work has been devoted to the simplest possible systems, i.e., to systems composed of objects with a hard core and a soft (permeable) shell [7,9]. Relatively fewer advances have been made in the more general and difficult case of interacting objects [11–14].

The problem of percolation in equilibrium systems composed of interacting objects is not just of theoretical interest since it is of immediate relevance to actual systems which became of wide interest in recent years. Conspicuous examples are the physics of water [15], of molecular liquids [16], of microemulsions [11,17], of deposition processes [18], and of polymerization [19].

The principal route for testing theoretical percolation models or for interpreting related experimental observations is computer simulations [1,9]. Thus far, very few computer simulations have been reported for the study of systems with interacting objects and those available were mainly lattice models [19–21]. These include a number of models [22,23] on percolation in correlated sequential adsorption (CSA) which is somewhat reminiscent of our work. However, like other lattice models, CSA does not relate directly to a physical microscopic interaction but rather to a more or less arbitrary enhancement of adsorption rates. This contrasts with the present work, in which we use the actual physical interobject interaction in the system. For such more realistic continuum models only the Metropolis method has been utilized [11] up to now. However, as we argue later, this method is somewhat unnatural and not too efficient for percolation. We present then an alternative method for simulating interacting systems which is used here for obtaining their percolation thresholds. The method is based on the expectation (which will be discussed below) that in such systems each particle is subjected essentially to a central potential-like interaction. The analytic analysis which confirms this premise of the present method will be given elsewhere [24]. In this Rapid Communication we concentrate, however, on this presently suggested computational procedure since we believe that it is not only conceptually simpler but that it is more natural and more efficient for percolation problems in general.

Let us first recall briefly the essentials of the Metropolis algorithm [25,26]. The algorithm starts with a given density of objects which remains constant during the whole run. Starting from some initial configuration, new ones are generated by random trial changes. Every such trial change is either accepted or rejected according to a criterion based on the change in the total energy of the system produced by the trial move. Statistical averages of quantities are calculated by sampling such a series of configurations generated according to the Gibbs distribution. In percolation problems [11], this algorithm is used to calculate the average probability P for the existence of a spanning cluster in the system. One must then vary the initially fixed density and repeat the whole process until such a density is found for which P reaches a prechosen value [11]. We note that the Metropolis process cannot, as a practical matter, be implemented with a large number of objects (in Ref. [11], for example, about $N=500$ particles and 4×10^4 steps per particle have been used) and as a result, large fluctuations are expected.

Because the percolation transition occurs upon increasing object density there is something unnatural and impractical in the Metropolis requirement of a full run of the simulation for each different object concentration. A more natural method would be to change continuously the object density and wait for percolation to occur. This

random-adding method is commonly used for the case of zero interactions (permeable objects), with bonding defined by partial overlap of the objects [4–10]. The concentration is continually increased by randomly throwing more objects into the system until a spanning cluster is obtained. Thanks to an efficient algorithm developed by Hoshen and Kopelman [27], such a simulation [10] can be conveniently carried out with large samples (typically $N \gtrsim 10^4$). Since interactions are absent, any configuration is just as probable as any other, hence, any random configuration already represents the equilibrium state. This is obviously not the case when interactions are present. Moreover, it has been known [28] for many years that even in the simplest case of purely repulsive hard object interactions, random-adding algorithms do not generate truly equilibrium configurations. However, it is not clear whether a random-adding algorithm can reproduce the connectivity properties of interacting systems, i.e., their percolation thresholds. This is distinct from the more general question of whether or not random-adding algorithms reproduce all the equilibrium properties of the system. We claim in the present paper that a random-adding algorithm *can* be developed which reproduces at least the connectivity properties of the system. We present it first for the simplest case of interacting objects, i.e., objects of a hard core–soft shell type (excluded volume repulsion). One can use then the same method as with noninteracting objects, provided one adds a removal criterion: if a newly added object falls on the hard core of another object, it is removed and another one is randomly thrown in its place. Indeed, as we show below, in spite of the fact that such a random-adding procedure does not generate a true equilibrium configuration [28], we have obtained a striking agreement between our results and the results derived for the equilibrium case which were obtained by the Metropolis procedure [11]. Moreover, in this case our simple algorithm is very quick while, by comparison, the Metropolis procedure is very inefficient and uses a much larger amount of computer time.

In systems with finite interaction strength, some other criterion is needed. We suggest such a criterion here for the buildup of the system thus extending the above random-adding procedure to real systems where interactions prevail. We demonstrate the validity of our method for a system of spherical particles for which a short range square-well attraction potential exists. We have chosen this system since detailed Metropolis simulations have been presented [11] for it. We thus consider the potential:

$$u(r)=\begin{cases}\infty\ , & r<\sigma \\ -(kT)\epsilon\ , & \sigma<r<\sigma(1+\lambda) \\ 0\ , & \sigma(1+\lambda)<r\end{cases} \tag{1}$$

where $kT\ (\equiv 1/\beta)$ is the thermal energy, ϵ is a parameter proportional to the interaction's strength, $\sigma/2$ is the hard-core radius, and $\sigma\lambda$ is the width of the attractive well. The binding criterion is provided by adding a soft shell of diameter d: two spheres at a distance r are bound if $\sigma<r<d$.

Turning to our procedure let us choose a particle in the system and consider all the other particles in relation to it. We can take then the selected particle as the source of a potential $u(r)$ to which all other particles respond. If these particles were all independent they would arrange themselves around the selected particle according to the Boltzmann distribution. In such a case, the density $\rho(r)$ of particles at a distance r from the selected particle would behave as

$$\rho(r)\propto\exp[-\beta u(r)]\ . \tag{2}$$

We will argue below that Eq. (2) is a good approximation for systems such as the one we study. In that case, we can utilize it to take the interactions into account in the following way. Let us define V_W as the volume around a particle's center, into which the potential well extends, and V_O as the remaining available volume, i.e.,

$$V_W=(4\pi/3)\{[\sigma(1+\lambda)]^3-\sigma^3\}\ ,$$

and (3)

$$V_O=V-(4\pi/3)[\sigma(1+\lambda)]^3\ ,$$

where V is the total volume of the system. Next we define N_O as the average number of particles whose distance r from a given typical particle is larger than $\sigma(1+\lambda)$, and N_W as the average number of particles whose distance from a given typical particle is less than $\sigma(1+\lambda)$. Apart from removing cases of hard-core overlap we now further require that the densities of particles around a given particle verify the relation

$$(N_O/V_O)/(N_W/V_W)\leq\exp(-\epsilon)\ . \tag{4}$$

This is the crucial physical step in our algorithm, which means that the density within the "attraction ring" is higher than outside the ring by the Boltzmann factor. The motivation to assume that this will be a good description of the system is as follows. If only hard-core interaction between the particles prevails the number of objects interacting with a given object is limited by their repulsion. If we add an attractive interaction this effect is reduced, thus enabling the density in the "attraction ring" to approach that of the one expected from the above "central-force" picture. For the two dimensional system to be considered below this effect (which we have calculated analytically [24]) changes the right-hand side (rhs) of Eq. (4) from $A=1.6$ in the hard-core case ($\epsilon=0$) to $A\exp(-\epsilon)$, where A is in the range $1\leq A\leq 1.3$, when the presently used attraction potential is included. One should note that A is a function of the parameters λ and ϵ [24]. We have chosen, however, to write the rhs of Eq. (4) as $\exp(-\epsilon)$ because it is a straightforward expression with a well understood meaning, and since using $1.3\exp(-\epsilon)$ instead of $\exp(-\epsilon)$ in the simulations yielded minor variations which will be discussed elsewhere [24]. The reason for the $\leq$ sign in Eq. (4) is the buildup process of the simulation. Since the $=$ sign cannot be obtained in finite samples we should only consider the other alternative, i.e., that $(N_O/N_O)(N_W/V_W)\geq\exp(-\epsilon)$. This latter criterion can be rewritten as a higher bound on N_W/V_W in the form $N_W/V_W\leq(N_O/V_O)\exp(\epsilon)$. This

higher bound, however, grows with increasing interaction strength. Since our algorithm works by *random* additions, N_W/V_W tends naturally to be quite close to N_O/V_O. As a result, if the higher bound is high enough, it will have no influence on the system. Therefore, the stronger the interaction, the less effect would be obtained. To avoid such unwanted behavior, we need to use Eq. (4), which yields an interaction-sensitive *lower bound* on N_W/V_W.

We give here the basic steps of the algorithm used for the application of our method. First, a "particle" is thrown into the system, and a second one is added in such a way that it falls within the first one's attractive well. This is to ensure that N_W is not zero, since otherwise the next particle will always be rejected. At each step, a new object is randomly thrown into the system, and the total number of objects (a counter I) is increased by 1. The program then makes two sweeps of the sample and checks several criteria. In the first sweep, the algorithm checks whether the new particle's hard core overlaps the hard core of any other particle. This step is made more efficient by dividing the sample space into subregions and checking only such subregions where intersection is possible [10]. If such an overlap occurs, the particle is removed, I is decreased by 1 and a new particle thrown in. If, after the first sweep, the particle is (temporarily) accepted, a second sweep is made in which the algorithm checks the distance r between the new object and all other (previously thrown in) objects. For each such pair, if $\sigma < r < \sigma(1+\lambda)$, a counter K is incremented by 1. The quantities N_W and N_O are defined then by

$$N_W = [N_W(\text{previous}) + K]/I \ , \qquad N_O = [N_O(\text{previous}) + I - K - 1]/I \ . \tag{5}$$

The implementation of criterion (4) now takes the following form: If $(N_O/V_O)/(N_W/V_W) > \exp(-\epsilon)$, the new particle is rejected, I is reduced by 1 and N_W and N_O are returned to their previous values; if not, the new particle is accepted. Each time a newly added particle is accepted, the program checks for the onset of percolation. This is done by a continuum version [9,10] of the Hoshen-Kopelman [27] algorithm. The I value at which this onset takes place is the critical number of particles, N_c.

In order to check the validity of our procedure we compared its results with those obtained by the Metropolis procedure in Ref. [11]. We have thus used a system of spherical particles which interact according to Eq. (1) with a constant $\lambda = 0.1$, and for various interaction strengths ϵ. In order to facilitate the comparison we have studied the same quantities used in Ref. [11] as a function of the ratio of the hard-core diameter to the "whole particle" (or disk) diameter, i.e., $\eta = \sigma/d$. Hence for three dimensional (3D) systems we have computed the "total excluded volume" of the particles at the percolation threshold, $\rho_c 4\pi d^3/3$, where ρ_c is their density (N_c in our unit volume sample) at the threshold [9–11]. Similarly, for the two dimensional (2D) systems we have computed the "total excluded area" of the disks $\rho_c \pi d^2$. We carried out our simulations on samples of typically $N_c \approx 10^4$ particles.

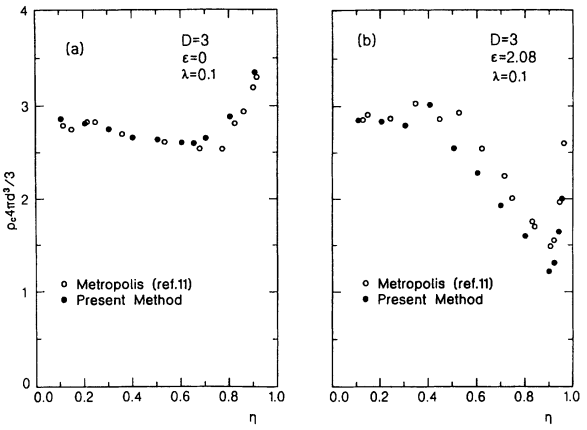

FIG. 1. Percolation thresholds as a function of the ratio η for a system of hard-core spheres (a), and for a system of interacting spheres (b).

The results for three dimensions, which are shown in Fig. 1, demonstrate an excellent agreement with the results obtained by the Metropolis procedure [11]. As pointed out above it is not *a priori* obvious [28] that for the hard core–soft shell case [Fig. 1(a)] the results should be much the same. Definitely in this case the Metropolis method is very inefficient in comparison with our random-adding procedure. It is, however, the excellent agreement in the case of a much more general interaction, shown in Fig. 1(b), which justifies our procedure. Since higher particle densities are involved in two dimensional systems [9,11] a more stringent test of the validity of the above procedure is obtained by performing the corresponding simulations. Indeed, the results shown in Fig. 2 reconfirm our procedure and prove our initial conjecture. Since the physical reasons and the particle clustering pictures which yield the behaviors shown in Figs. 1 and 2 have been clearly explained in Ref. [11], they will not be repeated here. We should finally remark that there are very slight discrepancies between our results and those of Ref. [11]. These discrepancies are probably no greater than those between different runs of simulations of identical systems with the same methods. Al-

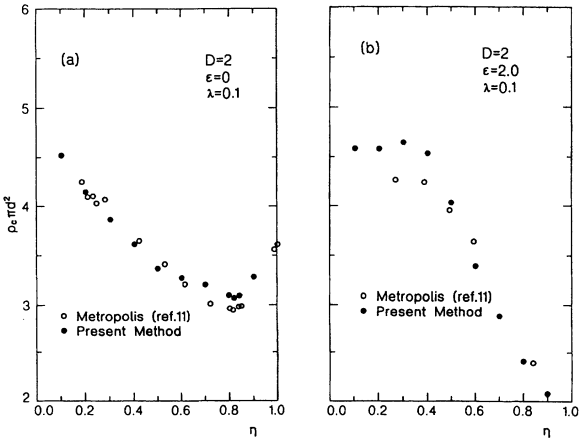

FIG. 2. Percolation thresholds as a function of the ratio η for a system of hard-core disks (a), and for a system of interacting disks (b).

though a better agreement can be obtained by using the numerical prefactors mentioned in connection with Eq. (4) we have chosen to demonstrate here the empirical validity of our very simple criterion [as given in Eq. (4)] because of its straightforward interpretation. Of course, in order to apply our procedure to a particular system, one has to check whether this criterion holds. We note, however, from the agreements shown in Figs. 1 and 2, that for the model suggested in Ref. [11] for microemulsions, our approximation is definitely applicable.

In conclusion, we have presented here a very simple method of simulation for computer-sample buildup in general, and for percolation problems in particular. The results of the method, which is more natural to percolation studies, are shown to reproduce the results of the Metropolis algorithm. This is in spite of the fact that the method does not necessarily reproduce actual equilibrium configurations. Nonetheless, it reproduces very well the percolation thresholds of the system. This strongly suggests that the connectivity properties of the system do *not* depend on all its equilibrium properties. We do not know why our simple method manages to capture the essentials of the percolation process in interacting systems, but the fact that it does points at some deep characteristics of the percolation transition and its relation to other thermodynamical properties of a system.

We thank A. Aharony, S. Alexander, and L. Kadanoff for very helpful discussions.

[1] D. Stauffer and A. Aharony, *Introduction to Percolation Theory* (Taylor and Francis, London, 1991).
[2] For a recent review see N. B. Isichenko, Rev. Mod. Phys. **64**, 961 (1992).
[3] For a review see I. Balberg, Philos. Mag. B **56**, 991 (1987).
[4] I. Balberg, Phys. Rev. B **33**, 3618 (1986).
[5] E. Cherlaix, E. Guyon, and S. Roux, Trans. Porous Media **2**, 31 (1987).
[6] A. S. Skal and B. I. Shklovskii, Fiz. Tekh. Poluprovodn. **7**, 1589 (1973) [Sov. Phys. Semiconductors **7**, 1058 (1974)].
[7] T. De Simone, R. M. Stratt, and S. Demoulini, Phys. Rev. Lett. **56**, 1440 (1986).
[8] U. Alon, A. Drory, and I. Balberg, Phys. Rev. B **42**, 4634 (1990).
[9] I. Balberg and N. Binenbaum, Phys. Rev. A **35**, 5174 (1987).
[10] A. Drory, I. Balberg, U. Alon, and B. Berkowitz, Phys. Rev. A **43**, 6604 (1991).
[11] A. L. R. Bug, S. A. Safran, G. S. Grest, and I. Webman, Phys. Rev. Lett. **55**, 1896 (1985); S. A. Safran, I. Webman, and G. S. Grest, Phys. Rev. A **32**, 506 (1985).
[12] S. C. Netemeyer and J. D. Glandt, J. Chem. Phys. **85**, 6054 (1986).
[13] J. Xu and G. Stell, J. Chem. Phys. **89**, 1101 (1988).
[14] U. Alon, I. Balberg, and A. Drory, Phys. Rev. Lett. **66**, 2829 (1991).
[15] J. Texeira, in *Correlations and Connectivity,* edited by H. E. Stanley and N. Ostrowsky (Kluwer Academic, Dordrecht, 1990), p. 167.
[16] D. H. Heyes and J. R. Melrose, J. Phys. A **21**, 4075 (1988).
[17] M. Lagues, J. Phys. (Paris) Lett. **40**, L331 (1979).
[18] M. C. Bartelt and J. W. Evans, Phys. Rev. B **46**, 12 675 (1992).
[19] A. Chhabra, D. Matthews-Morgan, D. P. Landau, and H. J. Herrmann, Phys. Rev. B **34**, 4796 (1986).
[20] R. Kikuchi, J. Chem. Phys. **53**, 2713 (1970).
[21] J. Kerterz, B. K. Chakrabarti, and J. A. M. S. Duarte, J. Phys. A **15**, L13 (1982).
[22] D. E. Sanders and J. W. Evans, Phys. Rev. A **38**, 4186 (1988); J. W. Evans, J. Phys. A **23**, L197 (1990).
[23] S. R. Anderson and F. Family, Phys. Rev. A **38**, 4198 (1988).
[24] A. Drory, I. Balberg, and B. Berkowitz (unpublished).
[25] See, for general exposition and some applications, *Monte Carlo Methods in Statistical Physics,* edited by K. Binder (Springer, Berlin, 1979), and references therein.
[26] Harvey Gould and Jan Tobochnik, *An Introduction to Computer Simulation Methods* (Addison-Wesley, Reading, MA, 1988), Chap. 16 and references therein.
[27] J. Hoshen and R. Kopelman, Phys. Rev. B **14**, 3438 (1976).
[28] B. Widom, J. Chem. Phys. **44**, 3888 (1966).