FRACTAL BEHAVIOR AND DYNAMICS ON PERCOLATING CLUSTERS

Panos Argyrakis
Department of Physics
University of Crete
Iraklion, Crete, Greece

ABSTRACT. We calculate very accurately several random walk properties on percolating clusters using various techniques. We derive the associated critical exponents characterizing fractal behavior and compare it to recent conjectures about their exact value. We extend all calculations to cover the whole range from the critical point to the perfect crystal, and thus observe the fractal-to-Euclidean crossover. We find that Euclidean behavior is achieved rather fast above the threshold point. We also investigate correlated motion on fractals, for which we find that it does not belong to same universality class as regular random walk does. Finally we look at long-range interaction clusters, for which we find that random walk is of similar nature as in nearest neighbor clusters.

1. INTRODUCTION

The theory and numerous applications of random walks in a variety of fields ranging from molecular/solid state physics to polymers to biological proteins and many more have attracted a continuous interest over the years, plausibly due to their success in explaining the corresponding phenomena studied [1]. The first pioneering work of Montroll [2], who systematically studied their properties by introducing the generating function method, was improved and refined by Henyey and Seshadri [3] and by Blumen and Zumofen [4]. Because of this we now have available closed form solutions for the number of sites visited in an N-step walk, S_N, the mean-square displacement, R_N^2, the probability for return to the origin, P_o (first passage time), etc. But all this work is concerned with perfect lattices or crystals, while it is recognized that the exact general solution for doped lattices for all concentrations is a formidable task. In this latter but very important problem it has also recently been recognized that a mean-field approach (an

R. Daudel et al. (eds.), Structure and Dynamics of Molecular Systems – II, 209–229.

effective medium approximation) may well provide adequate answers. Still it is expected that such an approach will suffice for small to medium dilutions only, while it will break down close to the critical percolation threshold. The region of the critical point has been a subject of intensive studies in the past, probably because of its connection to classical thermodynamics and phase transitions. But only recently with the advent of the notion of fractals [5,6] did it become apparent that random walk properties at the critical point can be evaluated exactly, something that threw new interest in the general problem discussed here. This is accomplished by introducing the fractal and fracton dimensionalities. These are fractional numbers (dimensions) smaller than the underbedding dimension, but their value gives an indicative measure of the disorder present, both for the structure of the disordered lattice and the dynamics on it. This prompted a surge of publications in a short period of time that discussed conjectures, hypotheses, numerical verifications, corrections to the scaling, etc.

In this paper we discuss our results of calculations that monitor diffusion on disordered lattices via random walks. We find accurate values for the fractal dimensionalities d_s by calculating S_N, the mean number of distinct sites visited in an N-step walk, and for the diffusion exponent D by calculating R^2_N, the mean-square displacement. We investigate in detail the effects of correlated diffusion and compare it to normal random walk. Finally we look into the effects of long-range diffusion. All these properties are calculated using a variety of algorithms (discussed in Section 2) first at the critical point, and then in the whole range above criticality up to a perfect lattice, so that we study in detail the crossover to Euclidean behavior.

It should also be noted that some efforts towards the solution of the general problem of random walks on disordered lattices were first published by us [7] using numerical solutions, and even though the model and methods were rather crude, they were still successful, at least partially, in explaining experimental data of luminescence from organic crystals at low temperatures. But it is only recently that more complete and satisfactory solutions are provided through the ideas of fractals.

2. METHOD OF CALCULATION

2.1. Technique

We use Monte-Carlo simulation methods to monitor several random walk properties. Our algorithm has been considerably improved in recent years, increasing both the

speed of operations and the size of the lattice used. Depending on the required application we utilize two techniques for the generation of random lattices. In the first case (a) we use the so called cluster-growth-technique, a method that generates and keeps in memory only the lattice portions used for the random walk. The lattice starts with one site only, and it is built continuously as the random walk proceeds by generating more sites adjacent to the diffusing particle. Once a site is gererated and its identity chosen it remains as such in the memory for the whole run. Using this method only one random walk can be executed on each lattice, which is more time consuming but has considerable advantages from the statistical point of view. The effective lattice size is now usually $4x10^6$ sites i.e. 2000x2000 for the 2-dim square lattice, and 160x160x160 for the 3-dim simple cubic. Consequently, the properties derived using this method pertain to walks that originate on any-size cluster, whether this may be the infinite percolating cluster or a monomer.

In an alternate approach, method (b), the whole lattice is generated and kept in memory before the beginning of the walk. This, although is a slower process, is necessary when it is required to find the critical percolating point exactly. In the previous case (a) the nominal critical occupational probability p_c was assigned, but in the actual numerical computation, due to statistical fluctuations, no exact realization of p_c is attained. After several realizations only its average value is attained. Thus, some realizations are well above p_c while others have not percolated at all. If the exact point is required it may turn out that this average quantity is not good enough, since close to criticality diffusion is not a linear process. Here in case (b) the lattice is initially generated at random well below the critical value, say at p=0.55 for the 2-dim square lattice. Then a certain number of closed sites is changed to open, their exact number and location being recorded. Usually this number is a power of 2, say 2^{10}. If after this change there is still no infinite cluster a new additional set of sites changes identity and the process is repeated until the critical point is surpassed. At this point the last set of sites is removed (i.e. changed back to closed sites), it is cut into two equal pieces and only the first half is now added; the lattice is tested again for criticality, but now with 2^9 sites changing identity. This process continues with the repeated dichotomy of the original number (2^{10}), until it goes down to 2^0. At this point we are assured that we are exactly at the critical point, i.e. one single site has caused the appearance of the percolating cluster. Testing for criticality is done using a new version of the Cluster-Multiple-Labelling-Technique (CMLT). The details of this

version are explained in the Appendix. We only need to apply the CMLT as many times as the power of 2, i.e. here we apply it 10 times, something not very time consuming. Using this technique we employ lattices of 2.5×10^5 sites, i.e. 500x500 for the square 2-dim case.

2.2. Computer language

It is commonly agreed that for Monte-Carlo simulations of the type reported here Fortran is not the best available language, since Fortran is best suited for numerical calculations, while the present work involves mainly integer manipulations and conditional statements. Another high level language (and easy to learn) that it better suited is C language. Of course, one would benefit the most by going to a low-level language, such as the assembler that each machine has. However, if it is necessary to use Fortran there are several points worth mentioning which if properly used can provide an added advantage. Since only five alternative pieces of information are needed at each time (the identity of a site, open or closed, and whether a site has been visited before or not, or whether it has not been defined), we can utilize more efficiently the length of each computer word by breaking each 32 bit word in four 8 bit sections. We now store the information for four sites in one word and effectively increase the size of available memory by a factor of 4. In Fortran this is done by use of a subroutine with the main array in Logical*1 and Integer*4 variables declaration, occupying the same memory space and continuously identified via an Equivalence statement. We thus avoid the difficult task of byte manipulation in machine language.

All different topologies in 1-dim, 2-dim, and 3-dim lattices are reduced to a one-dimensional array in the memory, so that it is not necessary to reach into the virtual memory as frequently. Thus the number of page-faults and transfers is decreased, and the overall speed is increased.

All work presented here was performed in a mini-computer, VAX 11/750 by Digital Equipment Corporation, with 4 Mbytes of direct memory and 550 Mbytes of virtual memory, and with the VMS operating system.

3. RESULTS

3.1. Spectral dimension

It is well established by now that the average number of distinct sites visited in an N-step walk behaves as:

$$S_N \sim N^{d_s/2} \quad . \tag{1}$$

Here d_s is the scaling exponent called the spectral (fracton) dimension, and it was the initial hypothesis [6] that d_s should have a universal value of $d_s=4/3$ for all lattices of all dimensionalities. But it was later conjectured [8] that this rule breaks down for $d\leq 2$ (where d is the Euclidean underbedding dimensionality), but is still valid for $d>2$. We will look into these assumptions carefully to check whether they can be verified.

Depending on what clusters does one use for the point of origin the spectral dimension of Eq. 1 will have different values. We use the notation d_s for case (b) of the previous section, i.e. for runs that can originate only on the largest percolating cluster. For case (a), i.e. for runs on any-size cluster we will have an analogous equation:

$$S_N \sim N^{d_s'/2} \tag{2}$$

But it has been shown [9] that d_s and d_s' are related through:

$$d_s' = d_s\left(2 - \frac{d}{d - \beta/\nu}\right) \tag{3}$$

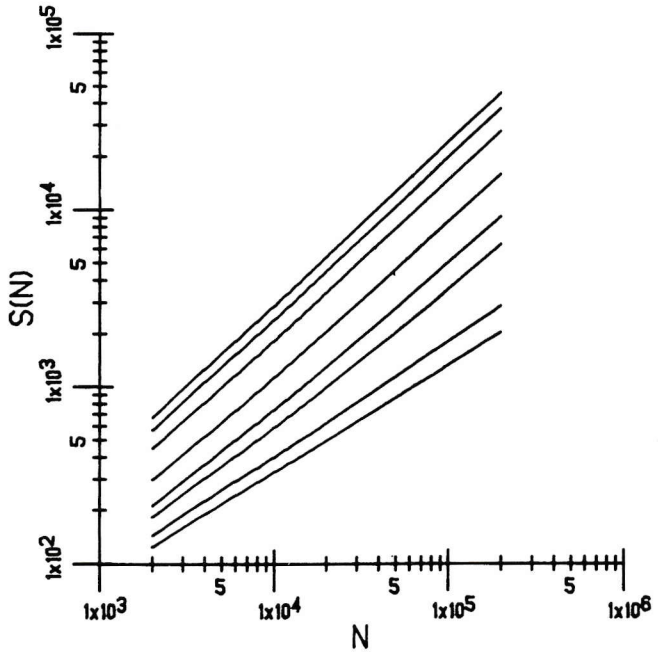

Figure 1: The number of sites visited S_N as a function of the number of steps N for 2-dim lattices. These are averages of 1000 realizations for walks that originate on any-size cluster (method a). Bottom to top: p=0.5931, 0.60, 0.63, 0.65 0.70, 0.80, 0.90, and 1.00.

where β and ν are the static percolation exponents. When one of the d_s or d_s' is calculated the other one can easily be deduced. Figure 1 shows a plot of $\ln S_N$ vs. $\ln N$ for several different occupational probabilities p for the 2-dim lattice, and Figure 2 the same plot for 3-dim lattice. The lowest curve in each case pertains to the critical percolating threshold and from its slope we receive [10]:

$$d_s' = 1.23 \pm 0.02 \quad \text{(2-dim)} \tag{4}$$

$$d_s' = 1.06 \pm 0.02 \quad \text{(3-dim)} \tag{5}$$

Using Eq. 3 and the values of Eq. 4 and 5 we get:

$$d_s = 1.30 \pm 0.02 \quad \text{(2-dim)} \tag{6}$$

$$d_s = 1.33 \pm 0.02 \quad \text{(3-dim)} \tag{7}$$

Using method (b), the method of the exactly percolating clusters, we calculate again S_N, but now for runs on the exactly incipient percolating cluster only. In Figure 3 we plot $\ln S_N$ vs. $\ln N$ for the 2-dim lattice. We observe that the data is almost fit on a straight line, but there are some deviations in the early time part. To avoid any such

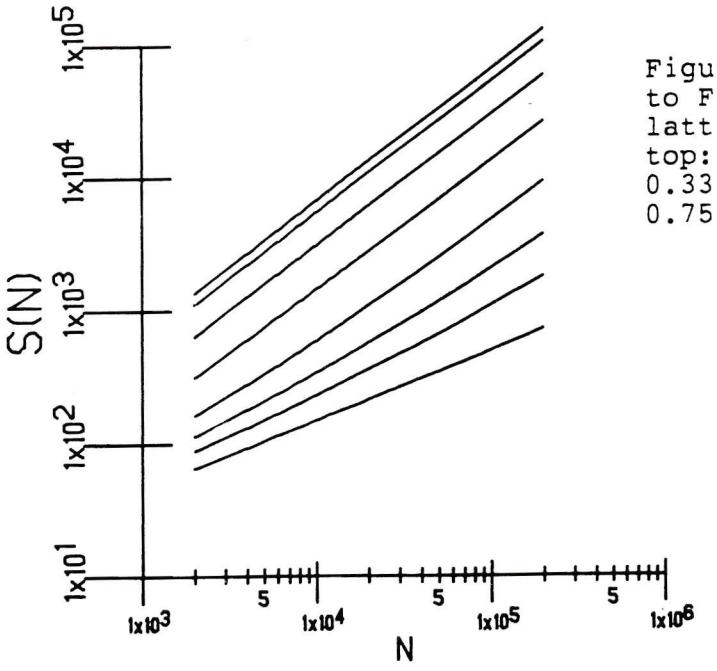

Figure 2: Plot similar to Fig. 1 but for 3-dim lattices. Bottom to top: p=0.3117, 0.32, 0.33 0.35, 0.40, 0.50, 0.75, and 1.00.

complications we notice from Eq. 1 that $S_N/N^{d_s/2}$ should be constant in time. We plot this quantity as a function of N in Fig. 4. Not being sure of the exact value of d_s we treat it as an adjustable parameter, and we plot the range

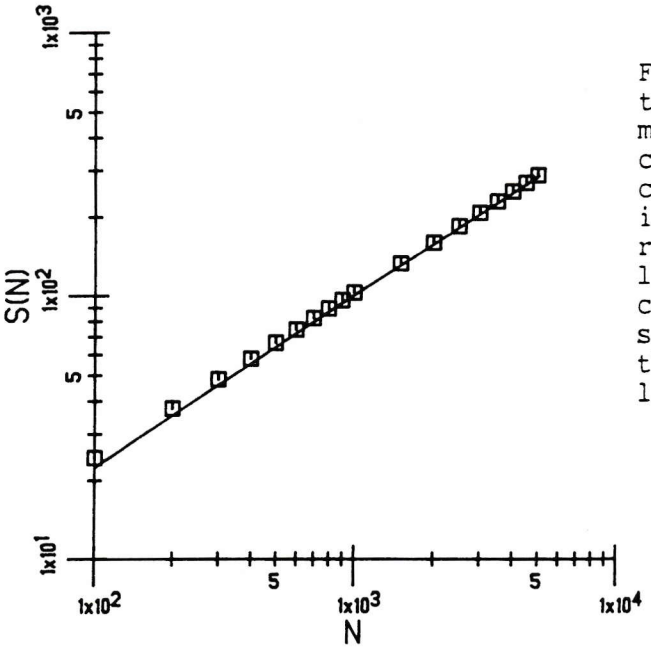

Figure 3: Plot similar to Figure 1 but using method (b). The calculation is at the critical point exactly, i.e. p=0.5931. These results are averages of 10000 realizations. The continuous line is a straight line to show the deviation from linearity.

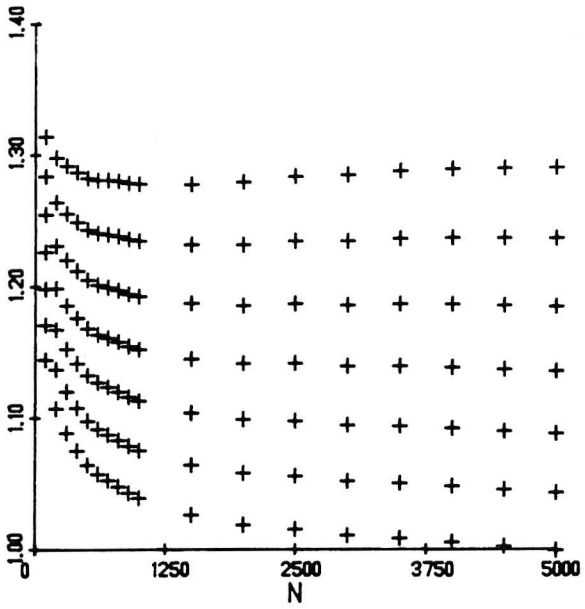

Figure 4: Plot of $S_N/N^{d_s/2}$ vs. N of the same data as in the previous figure. Here: d =1.27, 1.28, 1.29, 1.30, 1.31, 1.32, and 1.33 (top to bottom).

d_s=1.27 - 1.33. We observe that the values of d_s=1.29 or 1.30 are the ones that produce lines independent of time (again, aside from the early time part). Of course, if we use Eq. 3 to calculate d_s' we will receive: d_s'=1.23. We see that the direct calculation of d_s gives a value in agreement with the previous method [11].

Thus, in calculating the spectral dimension we employed two different approaches, methods (a) and (b), and they both produce the same result, i.e. for 2-dim lattices the proper value of the spectral dimension is d_s=1.30±0.02, in agreement with the Aharony-Stauffer [8] prediction, and a deviation of about 2% from the Alexander-Orbach-Rammal-Toulouse [6] theory. In 3-dim lattices d_s=1.33, as originally proposed.

3.2. The diffusion exponent

The mean-square displacement at the fractal limit behaves as:

$$R_N^2 \sim N^{2/D} \qquad (8)$$

Depending at the point of origin we again have two exponents from Eq. 8, D and D' (just like d_s and d_s'). Figure 5 shows $\ln R_N^2$ vs. $\ln N$ for several different p values for 2-dim lattices. From the slope of the lowest curve we derive a value for the D' exponent:

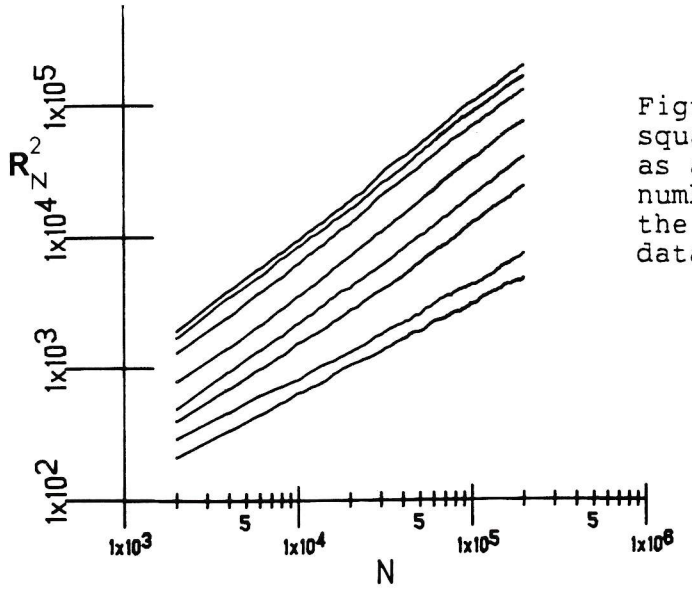

Figure 5: The mean-square displacement R_N^2 as a function of the number of steps N for the same p and the same data as in Fig. 1.

$$D' = 2.89 \pm 0.05 \qquad (9)$$

This value of D as calculated for the long time limit is in good agreement with previous work [12], but for small N (N=300 steps).

3.3. Crossover to Euclidean behavior

We can observe the crossover to Euclidean behavior in Fig. 1 by looking at the several curves in this figure. We notice that only the bottom and the top are straight lines. The bottom because it obeys Eq. 1 as it is in the fractal limit, the top because at p=1.00 there is an effectively simple power dependence, in spite of the well known logarithmic correction for the 2-dim walk. But for all intermediate p the slopes are varying and some lines are curved, with the curvature being a function of time, thus showing that each different p has a different effective spectral dimension that is time-dependent. The same behavior is observed at Fig. 2 for the 3-dim lattices. In order to see how fast does this crossover occur we plot the effective d_s, for the long time limit (N=200000 steps) as a function of the occupational probability p. The result is shown in Figure 6, for both the 2-dim and 3-dim lattices. One can see that for both cases the effective d_s sharply increases in the region

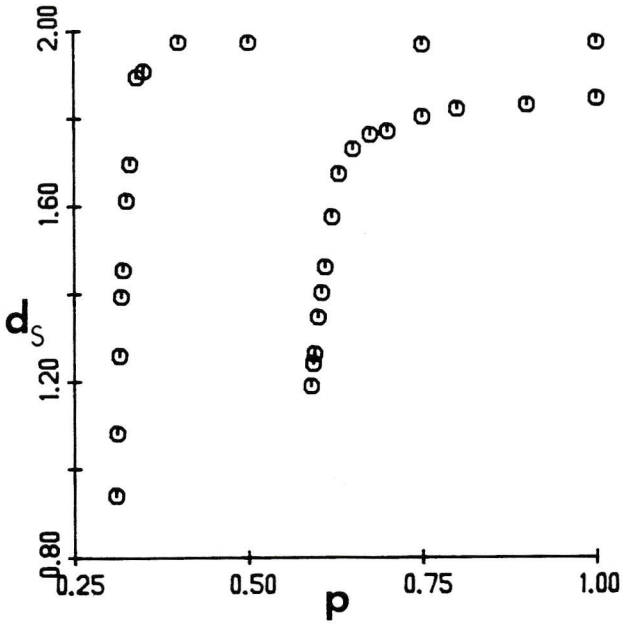

Figure 6: The effective d_s as a function of the occupation probability p, for 2-dim (lower) and 3-dim (upper) lattices, from the long time limit (N=200000 steps).

immediately above the critical point and it is approaching fast the classical value of 2. The limiting values here are: 1.98 for the 3-dim case, and 1.89 for the 2-dim case (a discrepancy from the classical value is expected here due to the logarithmic correction term that the 2-dim formalism contains [2]). From the curvature of the lines in Fig. 1 and 2 we would expect that the shape of the lines in Fig. 6 is time dependent. We found this to be true, however, we also observed that at any time period the main feature of the sharp rise remains intact.

As a consequence of the crossover behavior we expect that scaling will be valid only in a small region close to the threshold point. The scaling relationship is [13]:

$$S_N = N^{d_s/2} \, f\left[(p/p_c - 1)\, N^{1/(2\nu-\beta+\mu)}\right] \qquad (10)$$

where ν, β, and μ are the usual percolation exponents. In Fig. 7 we have a scaling plot for the 2-dim and 3-dim walks. We see that the different p values that fall within the scaling curve are all in the range: 0.31-0.35 (3-dim), and 0.59-0.65 (2-dim). Several time intervals are also included and all fall in the curves shown. Scaling breaks down for p>0.35 (3-dim) and for p>0.65 (2-dim), since as it can also be seen from Fig. 6 above these limits the Euclidean values are already attained.

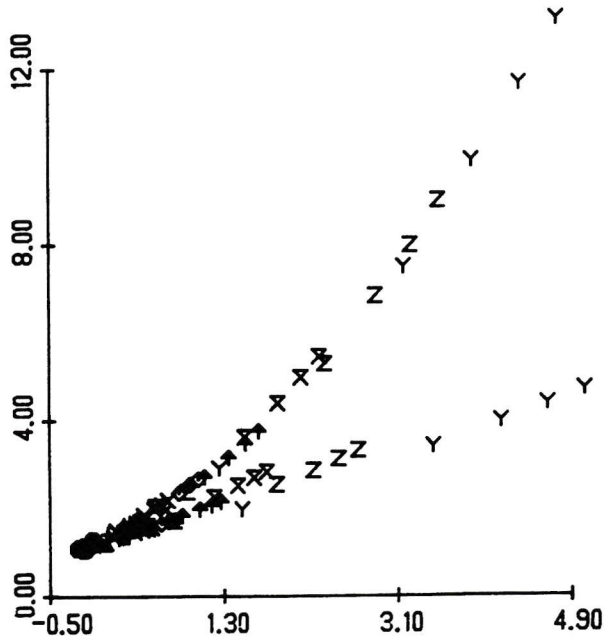

Figure 7: Scaling plot of $S/N^{(d_s/2)}$ vs. $(p/p_c - 1)\, N^{1/(2\nu-\beta+\mu)}$ 2-dim lattices (lower, 0.59<p<0.65), 3-dim (upper, 0.31<p<0.35), all for several time intervals.

3.4. Short-time corrections

As it is seen from Fig. 3 and 4 for small values of N, say up to N=1000 steps, Eq. 1 does not hold. In Fig. 3 linearity is not obtained for the entire range tested, while in Fig. 4 we see that the factor $S_N/N^{d_s/2}$ is not constant in time, but it is decreasing up to the value of about N=1000 steps. This leads us to believe that additional corrections to scaling are necessary in order to properly describe the early-time limit. Apparently, in early times the particle samples new lattice areas at a faster pace than after some time has elapsed, and some "state of equilibrium" is reached. This is also in agreement with previous assumptions [14,15]. The additional term is contained in the following:

$$S_N \sim N^{d_s/2} (1 + AN^{\omega}) \qquad (11)$$

where ω is the new exponent and is always necessarily negative, so that the contribution of the second term in parenthesis goes to zero for large N. Setting the constants of proportionality (a and b) in Eq. 11 and rearranging we receive:

$$S_N/N^{ds/2} - a = bN^{\omega} \qquad (12)$$

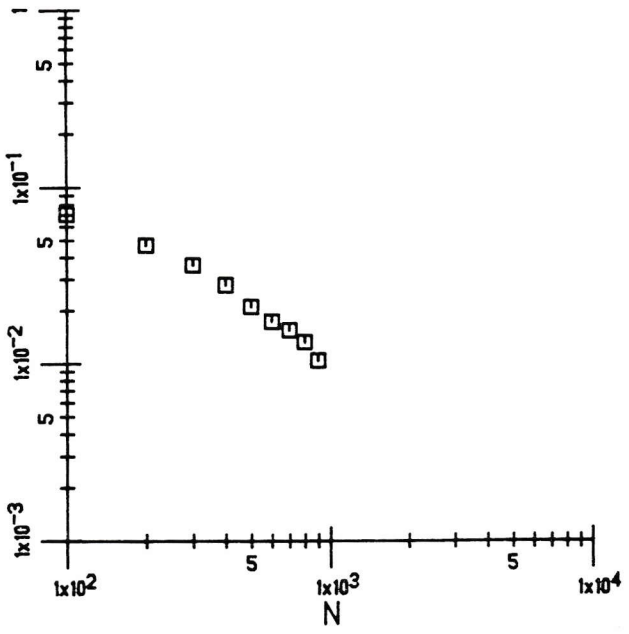

Figure 8: Plot of $S_N/N^{d_s/2}$ vs. N where d_s=1.29 and a=1.184 for the same data as in Fig. 3. The slope (giving the exponent ω) is taken from a straight line drawn at small N.

Here a represents the constant $S_N/N^{d_s/2}$ value for large N, which as seen from Fig. 4 has the value of a=1.18. Therefore, if we plot Eq. 12 directly in logarithmic form we will recover immediately the ω exponent. This is done in Fig. 8 using the same data as in Fig. 4. We consider the slope at short times because this is where the correction term predominates. At the straight line segment in Fig. 8 the slope is: ω=-0.48±0.08. We notice that forcing an exponent d_s=4/3 gives a=0.99 and ω=-0.47, showing that ω is not as sensitive to d_s as, obviously, a is.

3.5. Correlation effects

Correlation in diffusional motion has been shown in the past to be a necessary idea for the explanation of experimental data ranging from the diffusion of hydrogen in metals [16] and models of diffusion in concentrated lattice gases [17] to the relaxation mechanism of low-lying excited states of organic molecules at low temperatures [7] as studied by the use of random walk hopping models [18]. By correlation here we mean the retention of the directional memory over a certain number of lattice spacings. This is quantitatively described by the fraction p_f, which is the probability of a forward jump, and it is in the range: $1/a<p_f<1.00$, where a is the lattice coordination number. Recently we introduced [19,20] a new model that incorporates the effects of correlation in the usual [4] random walk models, first on perfect lattices [19], and then on mixed binary lattices [20]. Of interest to us here is whether correlated random walks behave the same way as regular walks and if the associated properties belong to the same universality class as the walks of the previous sections. For perfect lattices we extended Montroll's work [2] to include correlated motion. Thus, the formulae for S_N for uncorrelated walk are:

$$S_N = (8N/\pi)^{1/2} + \ldots\ldots \quad \text{(1-dim)} \tag{13}$$

$$S_N = \pi N/\ln N + \ldots\ldots \quad \text{(2-dim)} \tag{14}$$

$$S_N = cN + \ldots\ldots \quad \text{(3-dim)} \tag{15}$$

If we now include correlation by going through the generating function method and incorporating the correlation factor [19] we rederive these formulae as follows:

$$S_N = (8fN/\pi)^{1/2} + (1-f) + \ldots \quad \text{(1-dim)} \tag{16}$$

$$S_N = \frac{\pi f N}{\ln(8fN) + \pi(f-1)/2} + \ldots \qquad \text{(2-dim)} \qquad (17)$$

$$S_N = \frac{(1 - p_r) f N}{1+0.5(f-1) + (1-p_r)} + \ldots \qquad \text{(3-dim)} \qquad (18)$$

where f is the correlation factor (it is a function of p_f, but has different form for each dimensionality), and p_r is the return probability for uncorrelated walk. We see that we arrive at relatively simple modifications as compared to the uncorrelated walk model.

The problem is considerably more complicated in disordered lattices. Our results are given in Fig. 9. We evaluate the spectral dimension d_s' for correlated walks for several different values of the correlation parameter p_f, and for several different occupational probabilities p. The curve marked p_f =0.25 is the limiting case of no correlation at all, since in the square lattice all four directions carry the same 0.25 probability of scattering. It is also the same curve as that in Fig. 6.

We focus attention on the other curves in Fig. 9, which refer to higher p_f (p_f >0.25) values. In the fractal limit (p=0.60) we see that d_s' sharply decreases as p_f increases. For p_f =0.95 (at p=0.60) we see that d_s'=0.66 only. This sharp decrease is a consequence of the fact that at the critical percolation threshold correlated walks have a much

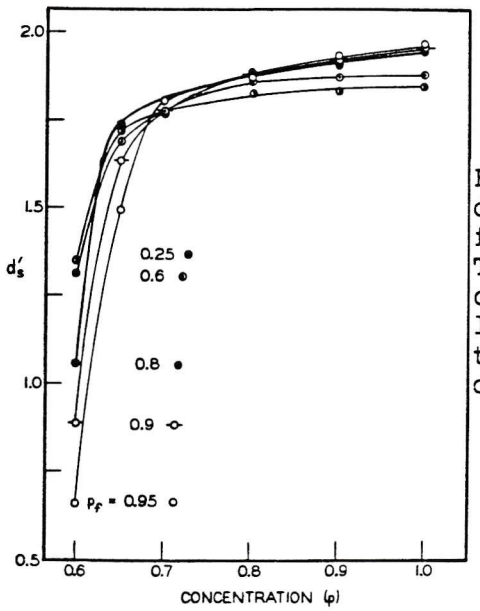

Figure 9: The spectral dimension d_s' vs. p, as a function of the forward correlation parameter p_f. The spectral dimensions are calculated in the long time limit, N=200000 steps, from walks that may originate on any-size cluster.

smaller S_N value than uncorrelated walks because, as was originally shown in the past [18], the particle indulges for long times in revisiting the same row of sites over and over again. We also used this idea [7] to interpret experimental data on mixed naphthalene alloys at 2 K.

As p increases, one observes for each p_f value the corresponding crossover to the classical behavior, since now, above p=0.80, correlated walks are much more efficient than uncorrelated ones. The region $0.70<p<0.80$, as seen from Fig. 9, is the "crossover region" between the different p_f values. At p=1.00 our calculations are in excellent agreement with formulae 16-18.

Thus, in this preliminary study for correlated random walk motion on fractal structures we investigated two types of crossover that occur, i.e. the crossover from fractal to Euclidean behavior for any type of walk, and the crossover from uncorrelated to correlated walk at any given p. We found that correlated random walks do not belong to the same universality class as simple walks. We notice that complete analytical solutions for the curves of Fig. 9 exist only for p=1.00, but more work is needed to quantitatively explain the behavior of the other curves.

3.6. Long-range interactions

It has been shown in the past [21] that the static critical percolation exponents β, γ, ν obey the universality hypothesis independent of the interaction range, for finite range cutoffs. It is interesting here to test the same hypothesis for the dynamic exponents examined here, i.e. the spectral dimension and corresponding crossovers. The interest stems from the well known experimental observation that triplet exciton transfer in organic molecules at low temperatures involves such long-range random hops on percolation clusters. The critical occupational probability p_c is a function of the interaction range, and it has been derived for various interaction ranges (R=1-5) by Monte-Carlo simulation [21] and by a position-space renormalization group approach [22]. Some of these p_c values are: $p_c=0.5931$ (R=1), $p_c=0.29$ (R=2), $p_c=0.16$ (R=3), $p_c=0.10$ (R=4), $p_c=0.07$ (R=5), $p_c=0.05$ (R=6), etc. We use a form: $P \sim e^{-\alpha r}$ for the stepping probability, where r is the distance, and α gives the shape of the curve for the distribution of distances.

The random walk process is monitored here in a similar way as in our previous calculations through S_N and R_N^2, with the only difference now that long steps are allowed according to the $P \sim e^{-\alpha r}$ equation. We find a complete analogy with the nearest neighbor case. Figure 10 is a plot of $\ln S_N$

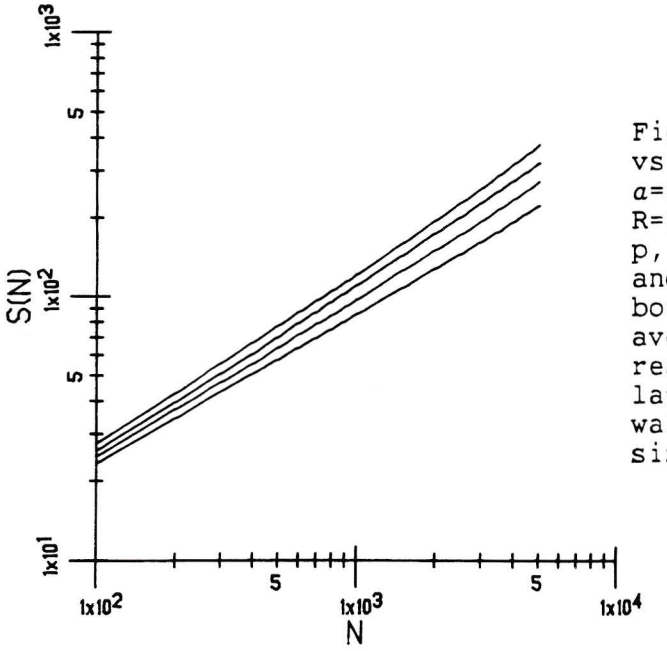

Figure 10: Plot of $\ln S_N$ vs. $\ln N$ for the case $a=0$, interaction range R=2, and four different p, p=0.32, 0.31, 0.30, and 0.29 (top to bottom). These are averages of 1000 realizations on 500x500 lattices. The random walks originate on any-size cluster.

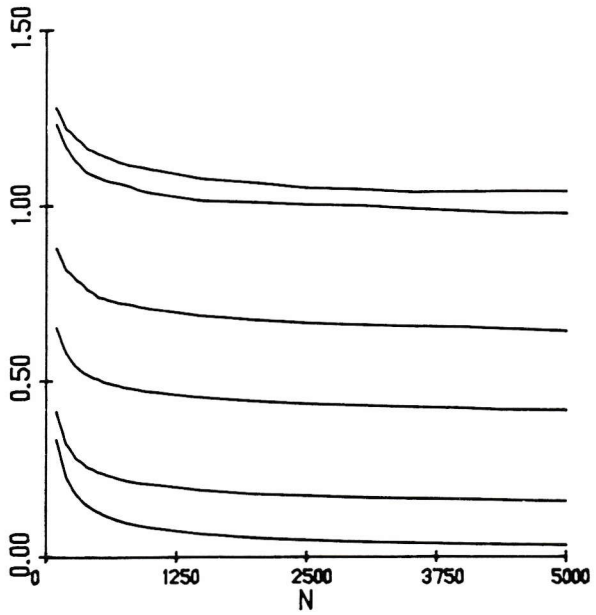

Figure 11: Plot of $S_N/N^{d_s/2}$ vs. N for different values of the parameter a. Here $a=0$, 0.5, 2.0, 3.0, 5.0, and 10.0 (top to bottom). Also, p=0.29 (range of interaction R=2). We used $d_s=1.23$.

vs. lnN, similar to Fig. 1, but for R=2 and we see that it yields a slope of d_s=1.24, same as Eq.4. To further investigate the scaling exponent we plot $S_N/N^{d_s/2}$ vs. N in Fig. 11 for different values of the parameter a in the range: $0<a<10.0$, for p_c=0.29 (interaction range R=2). We use d_s=1.23, and we see that for a=0 (all steps equally probable independent of distance) the quantity $S_N/N^{d_s/2}$ above N=2500 steps is constant in time. But for $a>0$, i.e. hopping probabilities that decrease with range, we see it takes longer (more steps) to reach the asymptotic limit. This is expected since it takes, on the average, several nominal time steps before an occurence of a long-range step.

We note that this model is based on defining and using a range (distance) dependent transition probability. But the actual form of this dependence (for example, the $e^{-\alpha r}$ used here) does not enter explicitly in the calculation. Any other proper form, such as for example r^{-n} (n=integer), could have been used with same results. One need only establish the proper correspondence. We conclude that long range random walks on percolating clusters behave similarly to simple random walks, with scaling and universality still intact. But for steeply falling-off step probabilities we find an asymptotic behavior that is approached more gradually in time. It would be interesting to check whether an experimental system (where the cutoff range itself is a function of time) will also exhibit an effective fractal dimension.

ACKNOWLEDGMENT. Parts of this work were performed with K.W.Kehr, A.Keramiotis, and R.Kopelman, whose collaboration is greatly appreciated.

APPENDIX

The cluster distribution is performed here using a new version of the Cluster-Multiple-Labelling-Technique (CMLT). The principal idea is the same as in CMLT, i.e. in cluster coalescence no sites need to be relabelled, and once a site is labelled, it retains its original label throughout the sweep of the lattice. The difference from the original method [23] is a new index processing used here. Instead of applying the routine CLASSIFY [23] at every site labelling in order to determine the proper cluster label of all neighboring sites we perform a single second sweep of the lattice that does the same operation. But this way, each site is checked only once in the second sweep, instead of twice that routine CLASSIFY does, for the square lattice topology, where every site is a neighbor to two different sites. The algorithm is given below:

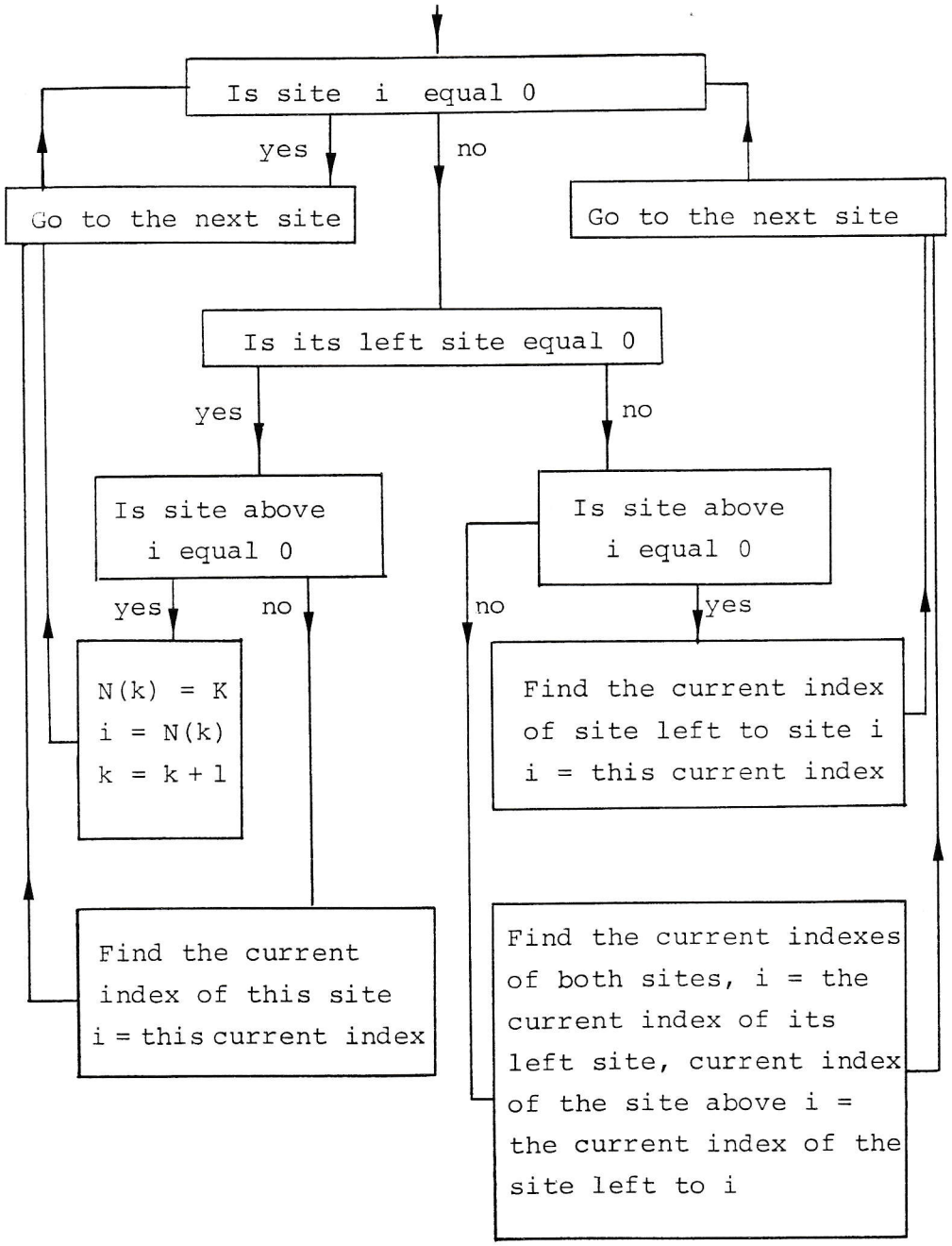

Figure 12: Flow chart for the site labelling assignment, for the first lattice sweep.

Occupied (open) sites are labelled 1 and empty (closed) sites are labelled 0. All labelling is done column by column starting at the top of the left-most column. We label each open site using a one-dimensional array N as follows: If both sites above and to the left of an open site *i* are closed sites, *i* takes the current index of the array N, which in the beginning is set equal to 1. If only one of these sites is open then *i* takes the current index of the open site, which, however, may be different from its original value. If both these sites are open then site *i* and the current index of the site situated above it simultaneously take the current index of the left site. This is all carried out in the one-dimensional array N, where the current index k (k=1, 2, 3, ..., etc.) of site *i* is the index which satisfies the equation N(k)=k. The value of the index for site *i* that will be stored in the array N is determined as follows: If the original index of site (*i*-1) is k_1 and the current index of site (*i*-1) is k_2, then we set for site *i* $N(k_1)=k_2$ and $N(k_2)=k_2$ so that k_2 is now the stored index for this site. The flow chart for this algorithm is given in Figure 12. When all index assignment in the lattice is finished we go back for the second sweep to renumber all open sites which do not have the latest assigned index. We then assign all identical indexes to a new one-dimensional array Z, where we store the size and latest index of each cluster.

To make this process better understood we work out an example of index processing for a 20x20 square lattice as shown in Fig. 13. Here the assigned p is p=0.56. In part (a) we have the lattice as it is formed using 0 and 1, for closed and open sites, respectively. In part (b) we begin by labelling the open sites of the first column. Following the algorithm in Fig. 12 the first open site takes the current index of the array N (N(1)=1), the second and third open sites also take the value 1, which is the current index of the site above them. The fourth site takes the current index of the array N (N(2)=2), and so on. When we start labelling the second column the first open site takes the current index of its left site, which is 1, the next open site takes the value 5, while the current index of the array N is taking the same value. The first site in the third column takes the value 6. The next one takes the value 1, while N(6)=N(1). Thus, when we label the fourth column the first open site takes the value 5 because now the current index of its left site is 5 (N(6)=1, N(1)=5, N(5)=5). Using this method we finish labelling all open sites in the lattice. In part (c), in the second sweep we label again all open sites which do not have their latest index. We see that the percolating cluster has as its index the value of 50, and is made of 83 lattice sites, the second largest cluster has a size of 71 and lattice index of 22, and so on.

```
1  0  1  1  0  0  0  1  1  0  1  0  1  0  0  0  1  1  0  1
1  1  1  1  0  1  0  0  0  1  0  1  1  1  1  1  1  0  1  1
1  0  1  0  1  1  0  0  0  1  1  1  1  1  1  1  1  0  1  0
0  1  1  1  1  0  0  1  1  1  1  0  1  1  1  1  1  0  1  1
0  0  0  0  1  1  1  1  0  1  0  1  1  0  0  0  1  0  1  1
0  0  0  0  0  1  1  1  0  0  0  0  0  0  0  0  1  1  1  0
1  1  1  0  0  1  1  1  1  0  1  1  0  1  1  1  0  0  1  0
1  1  1  0  1  1  1  1  1  0  1  0  1  0  0  0  0  0  1  1
1  1  0  0  1  1  0  0  0  0  1  1  0  1  0  0  0  0  0  1
1  0  1  0  0  1  0  1  1  1  0  0  0  1  1  0  0  0  1  1
1  1  0  1  1  0  1  1  1  1  0  1  0  0  1  1  1  0  0  0
0  1  1  1  1  0  1  0  1  0  1  0  1  0  1  1  0  0  0  1
1  1  1  0  0  1  1  0  0  0  0  0  1  0  0  1  0  1  0  1
1  1  1  1  1  0  0  1  1  0  1  1  1  1  1  0  1  0  0  1
0  1  0  1  1  0  1  1  1  1  1  1  1  1  1  0  0  1  0  1
0  1  0  1  1  1  1  0  1  1  0  1  0  1  0  1  0  1  1  1
1  0  1  1  0  0  0  1  1  1  1  0  0  0  1  1  0  1  1  1
1  1  1  0  0  0  1  0  1  0  1  0  0  1  0  0  1  1  0  1
0  1  1  1  0  0  1  0  0  1  0  0  0  1  0  1  1  0  1  0
0  0  1  0  1  0  0  1  1  0  0  0  1  1  1  0  0  1  1  1
```

Part (a)

```
 1  0  6  5  0  0  0 18 18  0 26  0 33  0  0  0 43 31  0 52
 1  1  1  5  0 13  0  0  0 24  0 30 11 31 31 31 31  0 49 31
 1  0  1  0 10  5  0  0  0 24 11 11 11 31 31 31 31  0 49  0
 0  5  5  5  5  0  0 19 11 11 11  0 11 31 31 31 31  0 49 31
 0  0  0  0  5  5 11 11  0 11  0 31 31  0  0  0 31  0 49 31
 0  0  0  0  0  5 11 11  0  0  0  0  0  0  0  0 31 31 31  0
 2  2  3  0  0  5 11 11 11  0 27 27  0 37 37 37  0  0 31  0
 2  2  3  0 11 11 11 11 11  0 27  0 34  0  0  0  0  0 31 31
 2  2  0  0 11 11  0  0  0  0 27 27  0 38  0  0  0  0  0 31
 2  0  7  0  0 11  0 20 14 14  0  0  0 38 38  0  0  0 50 50
 2  2  0  9  4  0 15 14 14 14  0 32  0  0 38 38 38  0  0  0
 0  2  3  3  4  0 15  0 14  0 28  0 35  0 38 38  0  0  0 53
 3  3  3  0  0 14 14  0  0  0  0  0 35  0  0 38  0 46  0 53
 3  3  3  3  4  0  0 21  4  0 29 22 22 22 22  0 44  0  0 53
 0  3  0  3  4  0 16  4  4 22 22 22 22 22 22  0  0 47  0 53
 0  3  0  3  4  4  4  0  4 22  0 22  0 22  0 41  0 47 42 42
 4  0  8  4  0  0  0 22 22 22 22  0  0  0 40 40  0 47 42 42
 4  4  4  0  0  0 17  0 22  0 22  0  0 39  0  0 45 42  0 42
 0  4  4  4  0  0 17  0  0 25  0  0  0 39  0 42 42  0 51  0
 0  0  4  0 12  0  0 23 23  0  0  0 36 36 36  0  0 48 48 48
```

Part (b)

```
50  0 50 50  0  0  0 18 18  0 26  0 50  0  0  0 50 50  0 50
50 50 50 50  0 50  0  0  0 50  0 50 50 50 50 50 50  0 50 50
50  0 50  0 50 50  0  0  0 50 50 50 50 50 50 50 50  0 50  0
 0 50 50 50 50  0  0 50 50 50 50  0 50 50 50 50 50  0 50 50
 0  0  0  0 50 50 50 50  0 50  0 50 50  0  0  0 50  0 50 50
 0  0  0  0  0 50 50 50  0  0  0  0  0  0  0  0 50 50 50  0
22 22 22  0  0 50 50 50 50  0 27 27  0 37 37 37  0  0 50  0
22 22 22  0 50 50 50 50 50  0 27  0 34  0  0  0  0  0 50 50
22 22  0  0 50 50  0  0  0  0 27 27  0 38  0  0  0  0  0 50
22  0  7  0  0 50  0 14 14 14  0  0  0 38 38  0  0  0 50 50
22 22  0 22 22  0 14 14 14 14  0 32  0  0 38 38 38  0  0  0
 0 22 22 22 22  0 14  0 14  0 28  0 22  0 38 38  0  0  0 42
22 22 22  0  0 14 14  0  0  0  0  0 22  0  0 38  0 46  0 42
22 22 22 22 22  0  0 22 22  0 22 22 22 22 22  0 44  0  0 42
 0 22  0 22 22  0 22 22 22 22 22 22 22 22 22  0  0 42  0 42
 0 22  0 22 22 22 22  0 22 22  0 22  0 22  0 40  0 42 42 42
22  0 22 22  0  0  0 22 22 22 22  0  0  0 40 40  0 42 42 42
22 22 22  0  0  0 17  0 22  0 22  0  0 36  0  0 42 42  0 42
 0 22 22 22  0  0 17  0  0 25  0  0  0 36  0 42 42  0 48  0
 0  0 22  0 12  0  0 23 23  0  0  0 36 36 36  0  0 48 48 48
```

Part (c)

Figure 13: An example of a cluster distribution for a 20x20 lattice. Part(a) A binary, substitutionally random square lattice with an assigned p=0.56. Part (b)Index assignment after the first sweep. Part (c) Index assignment after the second sweep.

REFERENCES

1. For a collection of papers on random walks see: Proceedings of Conference on random walks, Washington DC, J. Stat. Phys., 36,519-916(1983).

2. E.W.Montroll, Proc. Symp. Appl. Math., 16,193(1964).

3. F.S.Henyey and V.Seshadri, J. Chem. Phys., 76,5530(1982).

4. A.Blumen and G.Zumofen, J. Chem. Phys., 75,892(1982); J. Chem. Phys., 76,3713(1982).

5. B.B.Mandelbrot, The Fractal Geomerty of Nature (Freeman, San Francisco, 1983).

6. S.Alexander and R.Orbach, J. Phys. Lett., 43,L625(1982); R.Rammal and G.Toulouse, J. Phys. Lett., 44,L13(1983).

7. P.Argyrakis and R.Kopelman, Chem. Phys., 57,29(1981); Chem. Phys., 78,251(1983).

8. A.Aharony and D.Stauffer, Phys. Rev. Lett., 52,2368(1984).

9. I.Webman, Phys. Rev. Lett., 52,220(1984).

10. P.Argyrakis and R.Kopelman, J. Chem. Phys., 81,1015(1984); J. Chem. Phys., In press..

11. A.Keramiotis, P.Argyrakis, and R.Kopelman, Phys. Rev. B, 31,4617(1985).

12. D.Ben-Avraham and S.Havlin, J. Phys. A 15,L691(1982); J. Phys. A 15,L311(1982).

13. S.Havlin, D.Ben-Avraham, and H.Sompolinski, Phys. Rev. 27A(1983)1730.

14. R.Rammal, J. Stat. Phys., 36,547(1984).

15. R.B.Pandey, D.Stauffer, A.Margolina, and J.G.Zabolitzky, J. Stat. Phys. 34,427(1984).

16. J.W.Haus and K.W.Kehr, Sol. Stat. Comm., 26,753(1978); V.Lottner, J.W.Haus, A.Heim, and K.W.Kehr, J. Phys. Chem. Sol., 40,557(1979); W.Gissler and H.Rother, Physica, 50,380(1970).

17. K.W.Kehr, J. Stat. Phys., 30,509(1983).

18. P.Argyrakis and R.Kopelman, J. Theo. Bio., 78,205(1978).

19. K.W.Kehr and P.Argyrakis, J. Chem. Phys., Submitted for publication.

20. P.Argyrakis and K.W.Kehr, To be published.

21. J.Hoshen, R.Kopelman, and E.M.Monberg, J. Stat. Phys., 19,219(1978).

22. M.Gouker and F.Family, Phys. Rev. B, 28,1449(1983).

23. J.Hoshen and R.Kopelman, Phys. Rev. B, 14,3438(1976).