diffusion-fundamentals.org

The Open-Access Journal for the Basic Principles of Diffusion Theory, Experiment and Application

Diffusion and polymers in fractal, disordered environments

Niklas Fricke, Johannes Bock, Wolfhard Janke

Institut für Theoretische Physik and Centre for Theoretical Sciences (NTZ),
Postfach 100 920, 04009 Leipzig, Germany
niklas.fricke@itp.uni-leipzig.de

Abstract

We numerically investigate random walks (RWs) and self-avoiding random walks (SAWs) on critical percolation clusters, basic models for diffusion and flexible polymers in disordered media. While this can be easily done for RWs using a simple enumeration method, it is difficult for long SAWs due to the long-range correlations. We employed a sophisticated algorithm that makes use of the self-similar structure of the critical clusters and allows exact enumeration of several thousand SAW steps. We also investigate a kinetic version of the SAW, the so-called kinetic growth (self-avoiding) walk (KGW), as well static averaging over all RW conformations, which describes the so-called ideal chain. For the KGW, we use a chain-growth Monte Carlo method which is inspired by the pruned-enriched Rosenbluth method. The four walk types are found to be affected in different ways by the fractal, disordered structure of the critical clusters. The simulations were carried out in two and three dimensions.

1. Introduction

Percolation clusters are a simple model for disordered media such as porous rocks or biological cells [1, 2]. It is easiest to consider a square or cubic lattice with some randomly removed sites ("defects"). Most intriguing is the case where the concentration of sites that are no defects is at the critical percolation threshold, so that a lattice-spanning ("percolating") cluster can just barely exist. Such critical percolation clusters (CPCs) are highly irregular on all length scales (see Fig. 1, left) and have a non-integer Haussdorff dimension that is smaller than the embedding Euclidean dimension. This fractal nature has interesting consequences for random walks (RWs) taking place on the clusters. This model, known as the "ant in the labyrinth" [3] is often used to characterize diffusion in disordered media [4]. While random walks on regular lattices spread with the square root of the number of steps, the scaling exponent of the root mean square displacement is known to be reduced on critical clusters:

$$\left[\sqrt{\langle R^2\rangle}\right] \sim N^{\nu'_{\mathrm{RW}}}, \quad \nu'_{\mathrm{RW}} < 1/2. \tag{1}$$

The square brackets denote the quenched disorder average over all percolating cluster configurations.

Self-avoiding walks (SAWs) on CPCs have also widely been studied. They are a simple model for polymers in disordered environments [5]. Here, too, the scaling behavior of the mean square end-to-end distance (the analogue to the mean square displacement) is altered due to the fractal disorder; see for instance [6–9]. Interestingly though, the exponent ν'_{SAW} on the critical cluster is larger than on regular lattices.

There is a fundamental difference between those two models besides the excluded volume constraint for the SAW, namely the different statistical ensembles considered. The RW is regarded as a kinetic process where the next position is chosen uniformly from all available neighbors. In a random environment, different trajectories therefore have different probabilities, which are given by the inverse product of the number of available neighbors in each step. The mean square displacement expressed as an average over all trajectories thus reads:

$$\langle R^2\rangle_{\mathrm{RW}} = \frac{\sum_t w_t R_t^2}{Z} = \frac{\sum_t \left(\prod_i m_i\right)^{-1} R_t^2}{\sum_t \left(\prod_i m_i\right)^{-1}}, \tag{2}$$

with m_i denoting the number of available neighbors at step i. For the SAW on the other hand, we consider a static equilibrium average, where each conformation (trajectory) contributes equally. As a kinetic process, this would be described by a random walker that is absorbed whenever it steps on a defect or visits any site twice.

To complete the picture and allow for a sounder comparison, we investigate not only the RW and the SAW but also a random walk with uniform trajectory weights, the so-called ideal chain (IC) [10], as well as a kinetic version of the SAW, the so-called kinetic growth walk (KGW) [11]. The IC can be regarded as a polymer without excluded volume or as a random walker that is absorbed whenever it hits a defect. Note that the RW and the IC are identical on lattices without defects and with constant coordination numbers. The conformational average for the KGW looks the same as for the RW (Eq. (2)), but visited sites are not counted as available neighbors. However, the analogy to the diffusive RW is not perfect, as the KGW has the slight oddity that the walker can get trapped (see Fig. 1, right), in which case it is absorbed. It is like an ant trying to find its way through a labyrinth using the clue to avoid taking the same path twice and giving up when it has no other choice. The definition of the four different models are summed up in Table 1 below.

Table 1: Summary of the four different walk types.

properties:	not self avoiding	self-avoiding
diffusion statistics (kinetic)	**RW** Step rule: choose next position from non-defect neighbors. Trajectory weights: $(\prod_i m_i)^{-1}$.	**KGW** Step rule: choose from non-defect and non-visited neighbors; absorbed if none available Trajectory weights: $(\prod_i m_i)^{-1}$
polymer statistics (static)	**IC** Step rule: choose from all neighbors; absorbed at defects. Trajectory weights: uniform.	**SAW** Step rule: choose from all neighbors; absorbed at defects and visited sites. Trajectory weights: uniform.

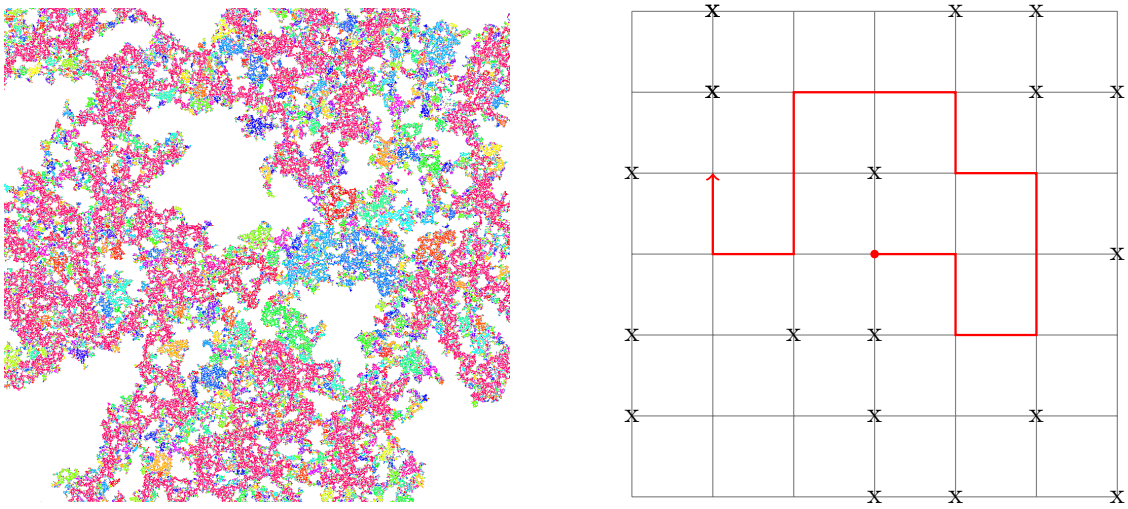

Figure 1: Left: Incipient cluster at percolation threshold. Colors show largest regions where each site is connected to at least two sites of the same region (bi-connected components), the largest being the cluster's backbone. Right: Kinetic growth walk (or SAW) that has got trapped.

2. Methods

To estimate the quenched average, we randomly create a large sample of CPCs and determine the average over all walk trajectories on each of them. The clusters are generated with the Leath method [12], which is based on a depth-first "burning" routine and works in the following way: Each lattice site can

either be occupied ("1"), empty ("-1"), or unknown ("0"). At the beginning, all sites are 0 except for one "seed" site, which is always 1. The burning starts at the seed, and a random number is drawn for each neighboring unknown site in turn. If the number is larger than the occupation probability p, the neighboring site is set to -1. Else it is set to 1, and the routine calls itself starting at the neighboring site. The program finishes once there are no more neighboring unknown sites, i.e., when all neighbors of the cluster are -1. A cluster is considered percolating ("incipient"), if it wraps around the system in at least one dimension (we assume periodic boundary conditions). For our simulations, only percolating clusters were used.

The challenging part is determining the averages over all trajectories of a given length. Here we used different numerical methods depending on the walk type. Those without self-avoidance (RW and IC) are easy to handle with exact enumeration [4, 13]. For the self-avoiding walks (SAW and KGW), this is difficult due to the long-term memory of the processes. We recently developed a method which exploits the fractal nature of the critical cluster and allows for very efficient exact enumeration of (normal) SAWs [14]. As there is no efficient way to enumerate the KGW trajectories, we had to resort to Monte Carlo simulations for this case. The method we used is much like the pruned-enriched Rosenbluth method (PERM) used for normal SAWs [15]. In the following, each method will be introduced briefly; more detailed descriptions can be found in the referenced works.

As a Markov process, the simple random walk can easily be studied with exact enumeration. This is because the probabilities for finding a walker on any site $\vec{r}_i$ after N steps can be calculated from the probabilities after $N-1$ steps:

$$P(\vec{r}_i, N) = \sum_n P(\vec{r}_n, N-1) W(\vec{r}_n \to \vec{r}_i),$$

where the sum goes over all neighbors of $\vec{r}_i$ and the transition probabilities, $W(\vec{r}_n \to \vec{r}_i)$, are given by the inverse of the number of available sites adjacent to $\vec{r}_n$. The method can also be understood in terms of Markov matrices acting on the probability state vector.

For the IC, the procedure is essentially the same, but we calculate the number of trajectories ending at each site instead of probabilities. The transition rates in this case are simply 1.

The straightforward enumeration method for SAWs is to use brute force and generate all possible conformations; see [7,16,17]. However, their number increases exponentially with the number of steps, and so does the computational effort. Fortunately though, the fractal structure of the CPCs offers a way to circumvent this problem [14]. The actual implementation of this method is rather complicated and will not be explained here, but the basic ideas are fairly simple. The key lies in the observation that the critical clusters are very weakly connected, so that they could be divided by cutting only a small number of bonds. Thanks to the self-similarity, this applies on all length scales. We can therefore partition the cluster into a hierarchy of nested "blobs" with very few interconnections (see Fig. 2) in order to factorize the enumeration:

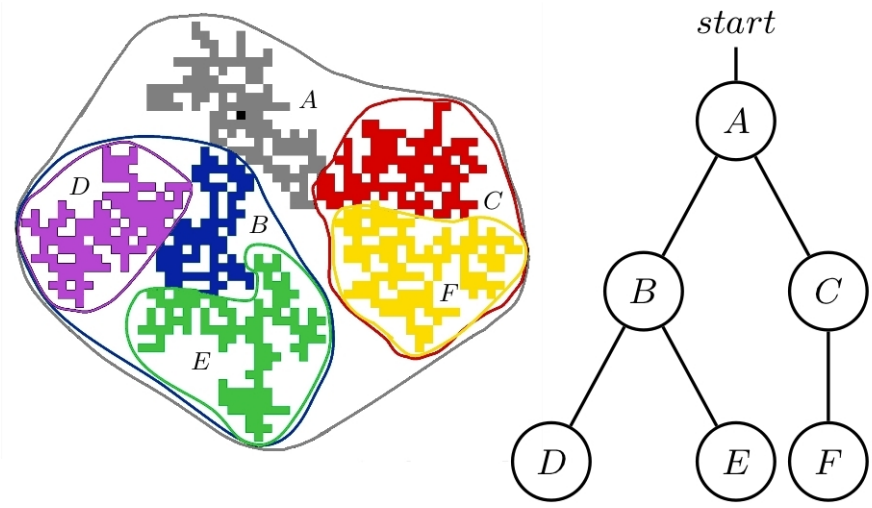

Figure 2: Decomposition of a critical percolation cluster into nested blobs and corresponding tree hierarchy. The SAW's starting position is marked black (in blob A).

We start by enumerating all SAW segments of any length through the smallest blobs and divide them into different classes, depending on how they connect the different entries to the blob. For instance, if a blob has two connections (a and b) to its parent in the hierarchy, we distinguish between the following classes: segments starting at a and terminating within the blob, segments starting at b and terminating in the blob, and segments connecting a and b. For the terminating segment classes, we also measure the mean end distance to the origin. We then "renormalize" the smallest blobs, treating them essentially as point-like while we repeat the procedure for the next larger blobs that contain them. The new segments have then to be matched with the right segment classes from before to determine their multiplicity and average end-point distances. This scheme is applied repeatedly going to ever larger blobs and ultimately the whole cluster. Correctly implemented, this method achieves polynomial (rather that exponential) increase of computation time with the number of steps, thus allowing for walks of several thousand steps. It can be used in any dimension, but we have only very recently generalized our implementation to more than 2D, so that the results on the 3D clusters are still very premature. The SAWs can also be simulated with PERM as was done for instance in [8, 9]. However, the new enumeration method is far more efficient; reliable results for more than a few hundred steps on a CPCs could probably not be produced with PERM [18].

The above method works, so far at least, only for normal SAWs and not for KGWs, but fortunately the latter can be treated quite easily and with reasonable efficiency using chain-growth Monte Carlo. In principle, all one needs to do is to randomly generate a large set of trajectories according to the step rule described before. The problem of this simple-sampling approach, however, is that the trappings lead to exponential attrition. Even though the attrition rate is much smaller than it would be for normal SAWs, it makes it hard to go to very long chains. To compensate it, we enrich the chain population by doubling chains during the growth from time to time. To keep the statistics correct, each chain gets a weight which is initially 1 and is halved each time a chain splits up. The splitting is triggered whenever the weight is larger than a constant $c_>$ times $\overline{W}$, the weight of all branches of the same length that have been generated so far divided by the number of started chains. We further increase the efficiency by stopping the chains from entering small pockets with no other exit ("dangling ends") where they would certainly perish. We therefore established an upper limit for the number of steps that can be taken in each pocket, and kept the chain from entering whenever this limit was smaller than the number of steps remaining to the desired length. However, avoiding such traps introduces a bias, which needs to be canceled by multiplying the weight with $1 - t/m$ where t/m is the fraction of available neighbor sites that lead into a trap. Since the weights can now shrink, we also introduce pruning: Whenever the weight is smaller than a (smaller) constant, $c_<$, times $\overline{W}$, it gets doubled but the branch is canceled with probability $1/2$. This avoiding of traps turned out to be crucial in the three-dimensional case. This method is very similar to PERM [15] for normal SAWs, where the weights are multiplied by a Rosenbluth factor of m in each step. However, the efficiency and reliability for this KGW-PERM version is much higher, so that 800 steps could conveniently be achieved. The reason that chain-growth methods perform better for KGWs is that they can only explore parts of the cluster, see Fig. 7 (top right). For the KGW, the omitted regions hardly contribute, but a very dense region that is difficult to explore can be critical to the SAW average.

3. Results

Before coming to the walks on CPCs, we shall briefly sum up the known results for regular lattices. As already mentioned, the RW and the IC are identical in this case and have the normal diffusive ($\sim \sqrt{N}$) behavior. For the SAW, ν is approximated by the Flory formula $\nu = 3/(2 + D)$, which happens to be exact in 2D. In 3D, there are very accurate numerical results [19]. For the KGW, numerical results are much less precise, but it is assumed that the exponent is asymptotically the same as for the SAW [20]. However, that issue has long been controversial as numerical studies had yielded smaller values [21]. The values are summarized in Table 2 below.

Table 2: Exponents for walks on regular lattices.

walk	RW	IC	SAW	KGW
ν_{2D}	1/2	1/2	3/4	$= \nu_{SAW}$ (Ref. [20]), 0.68 (Ref. [21])
ν_{3D}	1/2	1/2	0.587597(7) (Ref. [19])	$= \nu_{SAW}$ (Ref. [20]), 0.525 (Ref. [21])

The measured mean distances on CPCs are shown in Fig. 3. Each data point was obtained from

an independent sample of 10^4 randomly generated percolating clusters. The conformational averages, $\langle R^2 \rangle$, are exact for the RW, the IC, the SAW, and the first two data points for the KGW. The estimators for longer KGWs are still quite accurate, the error from the chain-growth Monte Carlo being much smaller than that on the disorder averages. As mentioned, the results for the 3D SAW are preliminary.

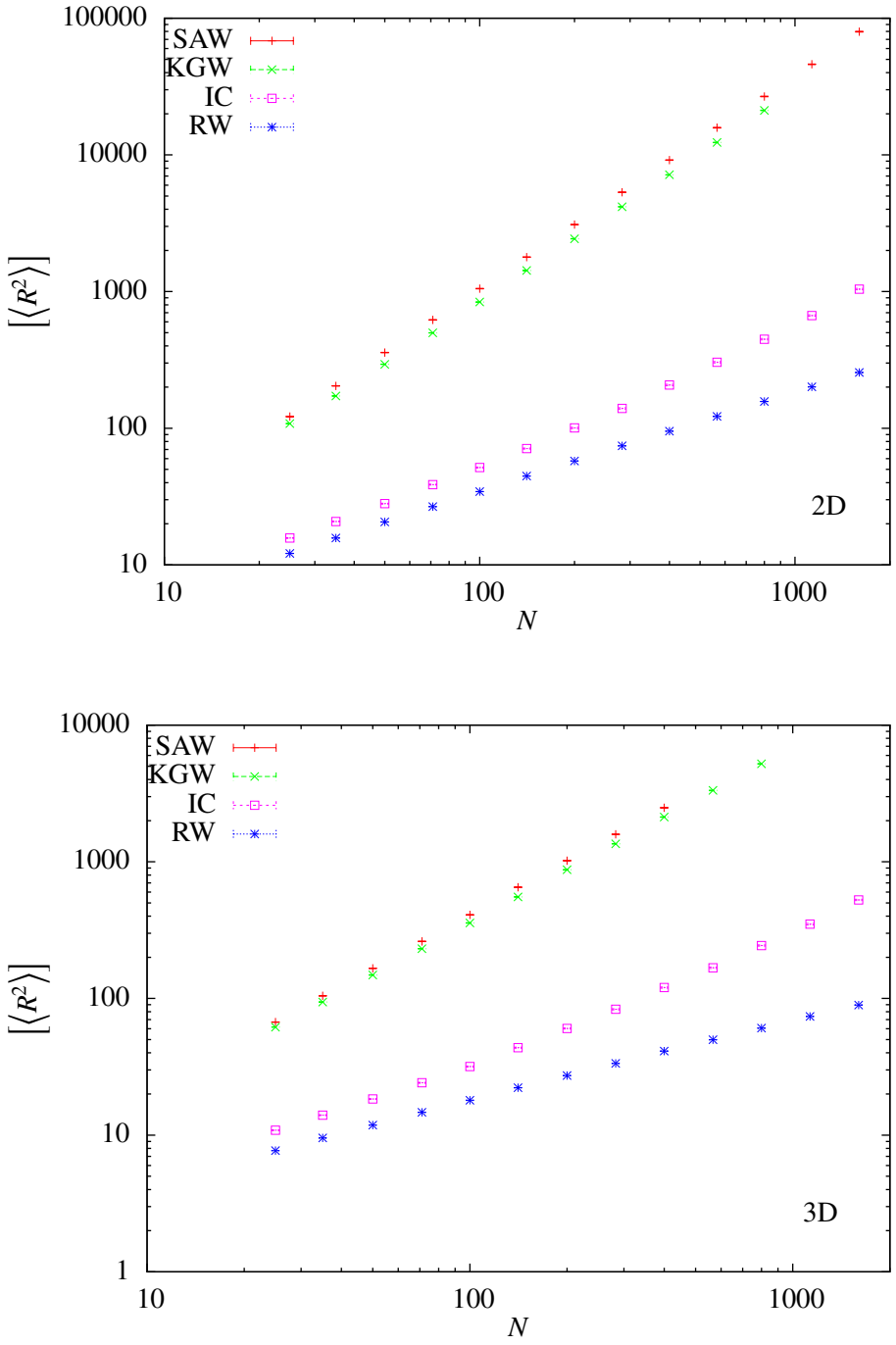

Figure 3: Log-log plots of the mean square end-to-end distance vs. the number of steps for the different walk types in 2D (top) and 3D (bottom).

Our estimates for ν presented in Table 3 are the insections with the y-axis of a linear extrapolations of $\frac{1}{2}\frac{\Delta \ln [\langle R^2 \rangle]}{\Delta \ln N}$ vs. $2/(N_i + N_{i+1})$; see Figs. 4 and 5. To reduce finite-size errors, only values for $N \geq 35$ were included in the fit. The statistical errors are due to the finite size of the sample of clusters for the disorder averages. For the RW, we find the sub-diffusive behavior ($\nu'_{\mathrm{RW}} < 1/2$) described in previous works. It should be noted that the exact enumeration method for the RW and the IC would allow for a

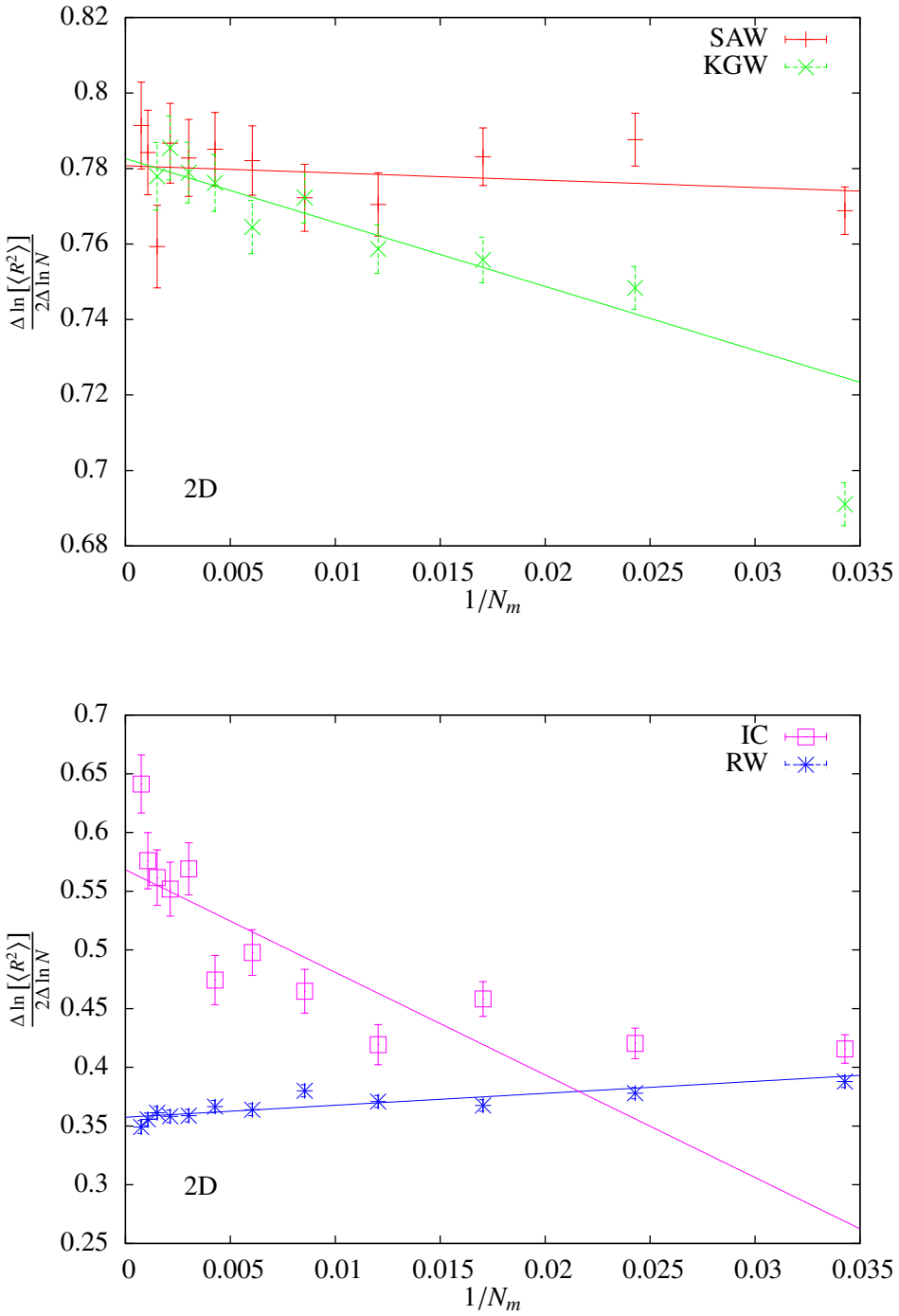

Figure 4: Extrapolations of $\frac{1}{2}\frac{\Delta \ln[\langle R^2 \rangle]}{\Delta \ln N}$ vs. $1/N_m$ to estimate the exponent $\nu \left(= \lim_{N\to\infty} \frac{d \ln[\langle R^2 \rangle]}{d \ln N} \right)$ in 2D. Only values where $1/N_m < 0.02$ were used for the fit, N_m denoting the mean of successive values of N.

significantly larger number of steps. However, we did not go beyond the length which is achievable for the SAW and the KGW, as the aim of this work is a comparison of the four models. The results for the IC look initially similar to those for the RW, but after about 200 steps (Fig. 3) the slope of the curve changes to a markedly larger value, which even exceeds 1/2 (super-diffusive behavior).

The shift to a larger value of the SAW exponent that has been reported in other studies is also observed for our results. For the KGW, for which a smaller ν had been found numerically on regular lattices, this shift appears to be more drastic, and the values for KGWs and SAWs seem to be identical

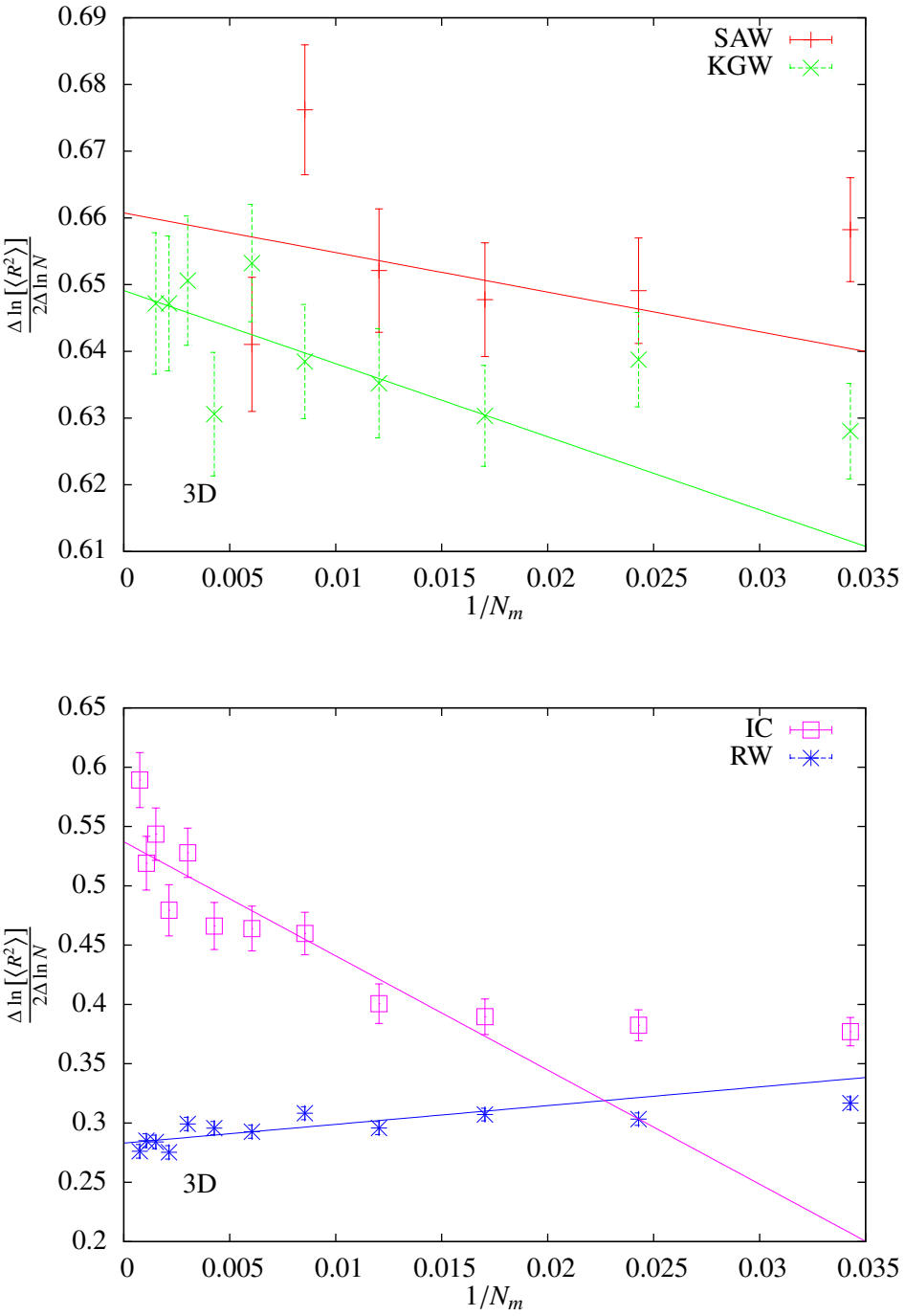

Figure 5: Extrapolations of $\frac{1}{2}\frac{\Delta \ln[\langle R^2 \rangle]}{\Delta \ln N}$ vs. $1/N$ to estimate the exponent ν in 3D. Only values where $1/N < 0.02$ were used for the fit, N_m denoting the mean of successive values of N.

Table 3: Results for the exponents on critical percolation clusters.

walk	RW	IC	SAW	KGW
ν'_{2D}	0.357(3)	0.57(2)	0.780(5)	0.782(3)
ν'_{3D}	0.283(4)	0.54(2)	0.66(3)	0.649(4)

within the limits of accuracy. This is very surprising since the two walks do actually behave very differently. This becomes evident when we look at the distributions of end-point locations and end-

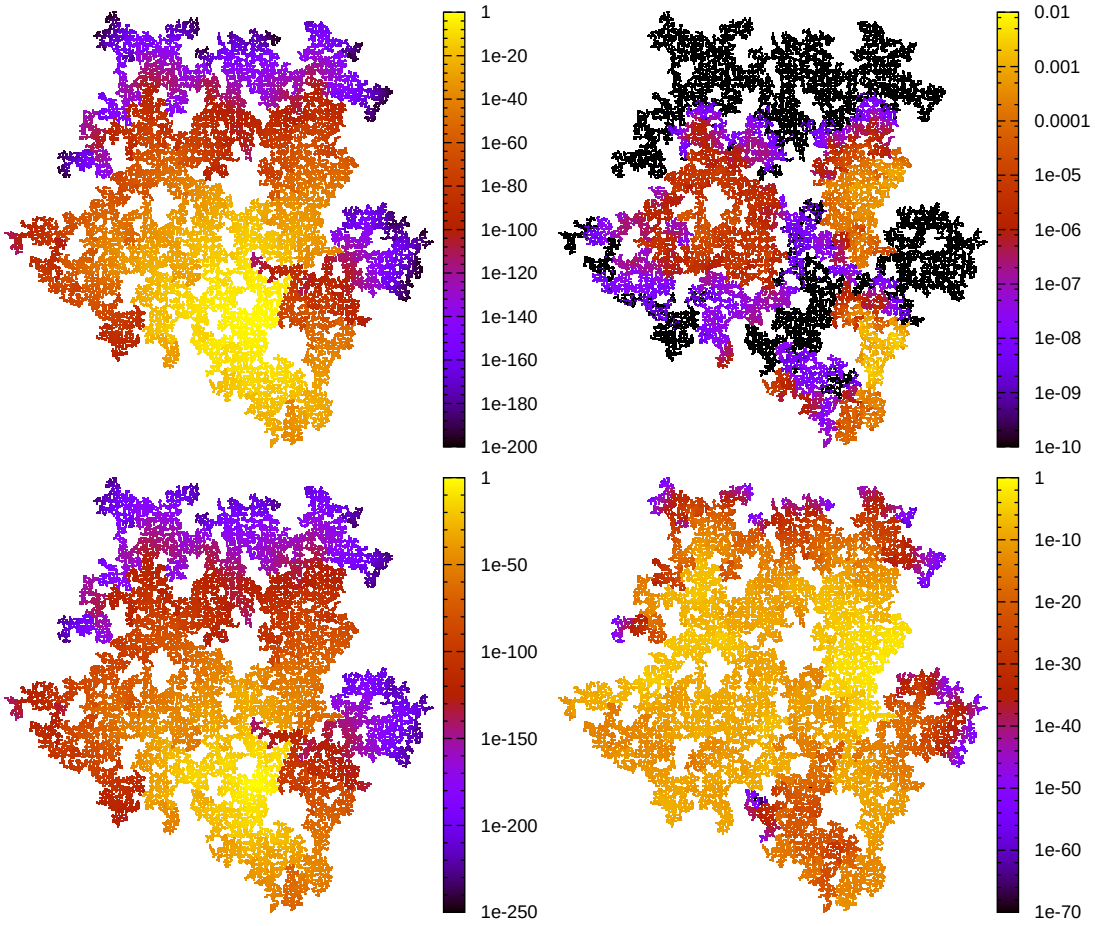

Figure 6: Distribution of end-point locations after 500 steps for the RW (top left), the KGW (top right), the IC (bottom left), and the SAW (bottom right) on a logarithmic color scale. The black areas have not been explored by the chain-growth Monte Carlo method for the KGW.

point distances. The shape of those distributions shown in Figs. 6 and 7 are typical: The end-locations of the RW and the KGW tend to be fairly spread out, while those for the IC and the SAW have very pronounced peaks. The reason for those peaks is that the conformational averages with uniform weights are usually dominated by very few cluster regions where the connectivity is highest. This probably also explains the crossover for the IC seen in Fig. 3: Initially, the region around the starting position dominates the entropy. Longer chains can explore a larger area and eventually a distant, denser region will become important and dominate until another, typically more distant region will take over. For the SAW the situation is similar, only that the self avoidance will actively push the ends out of regions closer to the origin as the chain length increases, see [18].

As a measure for this "localization" of the distributions, we looked at the average relative deviation of the mean square distance, $\left[\sigma_{R^2}/\left\langle R^2\right\rangle\right]$. As can be seen in Fig. 8, it seems to approach a constant value for the normal RW and drops fastest for the SAW and the IC.

4. Conclusion

We studied four different random walk types on critical percolation clusters: the standard random walk, the self-avoiding walk, the ideal chain and the kinetic growth walk. Some of the results confirmed previous studies while others, in particular those for the KGW, are novel or of unprecedented accuracy. (KGWs on CPCs have been studied before [22], but the author used an annealed averaging procedure.) The main outcome is that the scaling of the mean square displacement with the number of steps is increased for all but the RW model. This increase can be attributed to the stronger effect of self-avoidance in effectively smaller dimensions for the KGW, to the alteration of entropically dominant regions for the IC, and to a mixture of both effects for the SAW. Surprisingly, the exponents for the SAW and the KGW were found to be indistinguishable, even though the end-point distributions are quite

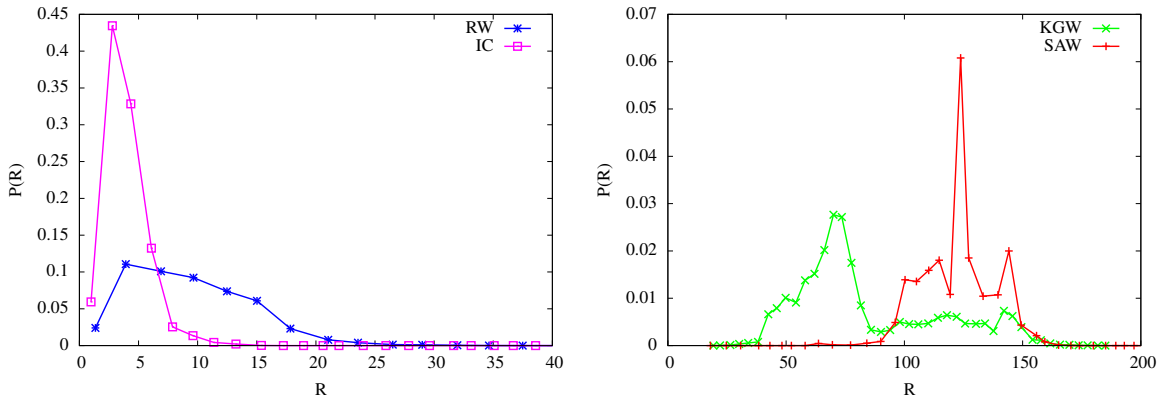

Figure 7: Probability distributions for end-point distances on the cluster shown in Fig. 6.

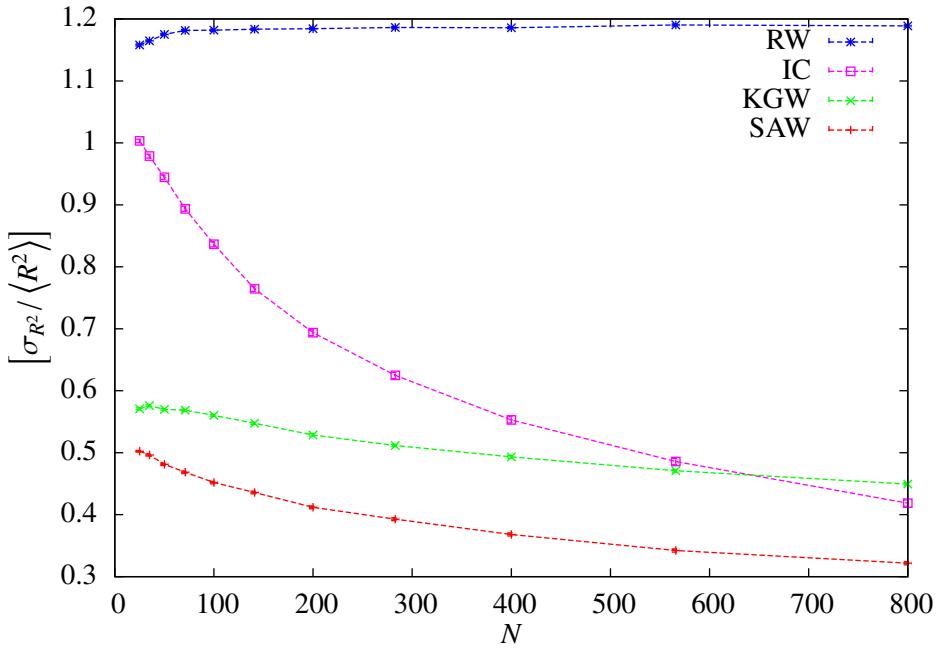

Figure 8: Average relative deviation of the mean square distance $\left[\sigma_{R^2}/\langle R^2\rangle\right]$ vs. N.

distinct. We also presented a compilation of the most effective numerical methods for the treatment of these models, among them a recently developed exact enumeration technique for SAWs and an efficient variant of PERM for KGWs on very dilute lattices.

Acknowledgments

This project was funded by the DFG via the Sächsische Forschergruppe FOR877. We are grateful for support from Leipzig Graduate School of Excellence "BuildMoNa" and the Deutsch-Französische Hochschule DFH-UFA under grant No. CDFA-02-07. We also thank Viktoria Blavatska for helpful discussions and the Alexander von Humboldt Foundation for facilitating these through their Institute Partnership Programme.

References

[1] C. Barton, P. R. La Pointe (Ed.): *Fractals in the Earth Sciences*. (Plenum Press, New York, 1995)

[2] R. Cuthbertson, M. Holcombe, R. Paton (Ed.): *Computation in Cellular and Molecular Biological Systems*. (World Scientific, Singapore, 1996)

[3] P.-G. de Gennes: *La percolation: un concept unificateur*. La Recherche **7**, 919-927 (1976)

[4] S. Havlin, D. Ben-Avraham: *Diffusion in disordered media*. Adv. Phys. **36**, 695-798 (1987)

[5] B. Barat, B. K. Chakrabarti: *Statistics of self-avoiding walks on random lattices*. Phys. Rep. **258**, 377-411 (1995)

[6] Y. Meir, A. B. Harris: *Self-avoiding walks on diluted networks*. Phys. Rev. Lett. **63**, 2819-2822 (1989)

[7] A. Ordemann, M. Porto, H. E. Roman, A. Bunde: *Multifractal behavior of linear polymers in disordered media*. Phys. Rev. E **61**, 6858-6865 (2000)

[8] V. Blavatska, W. Janke: *Scaling behavior of self-avoiding walks on percolation clusters*. Europhys. Lett. **82**, 66006-1-6 (2008); V. Blavatska, W. Janke: *Multifractality of self-avoiding walks on percolation clusters*. Phys. Rev. Lett. **101**, 125701-1-4 (2008)

[9] V. Blavatska, W. Janke: *Walking on fractals: Diffusion and self-avoiding walks on percolation clusters*. J. Phys. A **42**, 015001-1-18 (2009)

[10] A. Maritan: *Random walk and the ideal chain problem on self-similar structures*. Phys. Rev. Lett. **62**, 2845-2848 (1989)

[11] I. Majid, N. Jan, A. Coniglio, H. E. Stanley: *Kinetic growth walk: A new model for linear polymers*. Phys. Rev. Lett. **52**, 1257-1260 (1984)

[12] P. L. Leath: *Cluster size and boundary distribution near percolation threshold*. Phys. Rev. B **14**, 5046-5055 (1976)

[13] I. Majid, D. Ben-Avraham, H. E. Stanley: *Exact-enumeration approach to random walks on percolation clusters in two dimensions*. Phys. Rev. B **30**, 1626-1628 (1984)

[14] N. Fricke, W. Janke: *Scale-free enumeration of self-avoiding walks on critical percolation clusters*. Europhys. Lett. **99**, 56005-1-5 (2012)

[15] P. Grassberger: *Pruned-enriched Rosenbluth method: Simulations of* Θ *polymers of chain length up to 1 000 000*. Phys. Rev. E **56**, 3682-3693 (1997)

[16] P. M. Lam: *Exact series studies of self-avoiding walks on two-dimensional critical percolation clusters*. J. Phys. A **23**, L831-L836 (1990)

[17] A. R. Singh, D. Giri, S. Kumar: *Effects of molecular crowding on stretching of polymers in poor solvent*. Phys. Rev. E **79**, 051801-1-6 (2009)

[18] N. Fricke, W. Janke: *Self-avoiding walks on strongly diluted lattices: Chain-growth simulations vs. exact enumeration*. Eur. Phys. J. Special Topics **216**, 175-179 (2013)

[19] N. Clisby: *Accurate estimate of the critical exponent* ν *for self-avoiding walks via a fast implementation of the pivot algorithm*. Phys. Rev. Lett. **104**, 055702-1-4 (2010)

[20] A. Coniglio, N. Jan, I. Majid, H. E. Stanley: *Conformation of a polymer chain at the* Θ*' point: Connection to the external perimeter of a percolation cluster*. Phys. Rev. B **35**, 3617-3620 (1987)

[21] J. W. Lyklema, K. Kremer: *The growing self-avoiding walk*. J. Phys. A **17**, L691-L696 (1984)

[22] S. L. Narasimhan: *Kinetic self-avoiding walks on randomly diluted lattices at the percolation threshold*. Phys. Rev. E **53**, 1986-1989 (1996)