How to count 10^{100} self-avoiding walks on a critical percolation cluster

Niklas Fricke and Wolfhard Janke
Institut für Theoretische Physik, Universität Leipzig, Germany

Introduction

The scaling behavior of self-avoiding walks (SAWs) on critical percolation clusters has been widely studied since the early 1980s [2], being of interest for both theory and praxis (see [1]). Various Monte Carlo (MC) methods [3, 4] as well as complete enumeration [6, 7] were used to estimate observables of interest. However, neither approach proved very effective. While MC methods work quite well for weakly diluted lattices, they tend to become inefficient at criticality, and for exact enumeration the computational effort even scales exponentially with the number of steps.
We developed a new enumeration method which, inspired by the ideas of renormalization, overcomes this seemingly inherent exponential complexity by exploiting the structural properties of the percolation cluster. This new technique is restricted to systems at or close to criticality, but there it outperforms even the most sophisticated MC methods, while at the same time yielding exact results.

Self-avoiding walks

SAWs show universal scaling behavior, with "critical" exponents independent of system details. Most important are the exponents for the number of chains Z and the mean end-to-end distance $\langle R \rangle$:

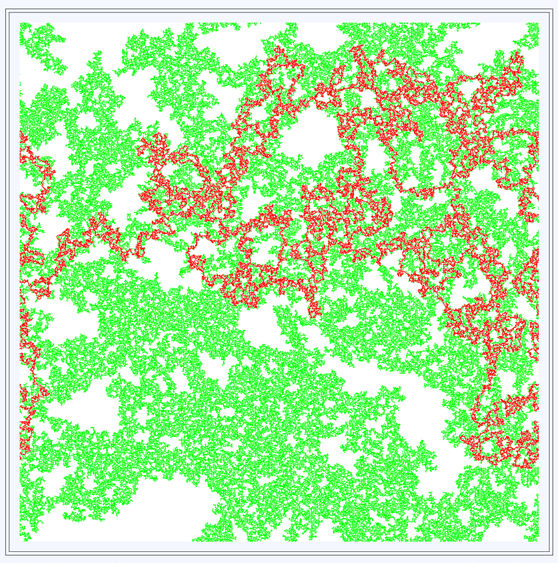
$$Z \sim \mu^N N^{\gamma-1}$$

$$\langle R \rangle \sim N^{\nu}$$

- Exponents for non-Euclidean substrates still controversial.
- SAW is simplest non-trivial model for a polymer.
- Extensions of the SAW include θ-polymers or the HP model.
- Percolation cluster is a crude model for a very crowded natural environment (e.g. biological cell).

Critical percolation clusters

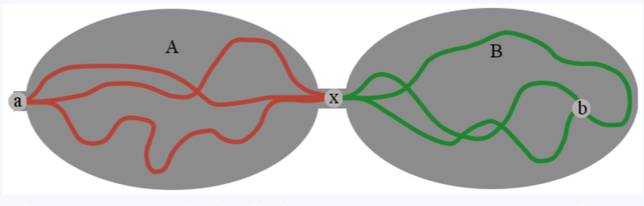
At the percolation threshold a cluster spanning the whole system is a fractal, i.e, it is self-similar and the scaling of its volume with system size is described by a non-integer exponent: $V \sim L^{d_f}$. It is just barely connected; removing only $O(1)$ would suffice to divide it. This is illustrated through the backbone (another fractal) which is what remains when the singly connected "pockets" are removed.



Percolation cluster on a 1000x1000 square lattice with backbone marked in red

Divide and conquer

If two parts of a cluster are connected through one site only, one can determine the paths on the whole cluster once the paths within the two parts are enumerated.



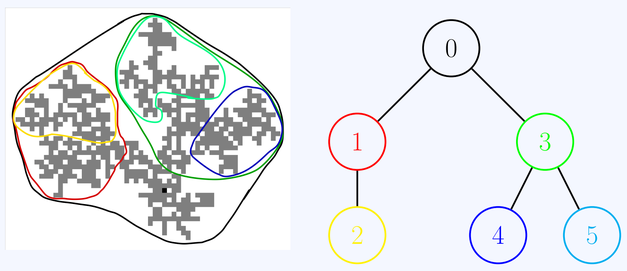
Paths $a \to b$ can be built from segments $a \to x$ and $x \to b$
The number of chains of length l from a to b is obtained via

$$Z(l)_{a,b} = \sum_{l_1,l_2=0}^{l} Z(l_1)_{a,x} \cdot Z(l_2)_{x,b} \cdot \delta_{l_1+l_2,l}$$

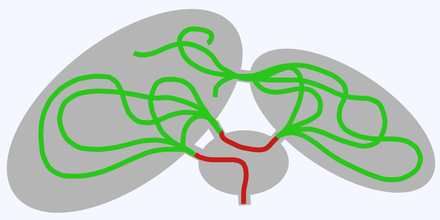
For more than one connection things are more complicated, but the principle remains the same.

Basic Strategy

- Create cluster and cut off remote regions.
- Decompose cluster into nested pieces ("blobs") with few connections on all length scales.



- Identify nesting of the blobs as hierarchical tree-structure, with the whole cluster as root.
- Determine paths though blobs whose "children" have been already dealt with, actively counting only in the region outside of the children.

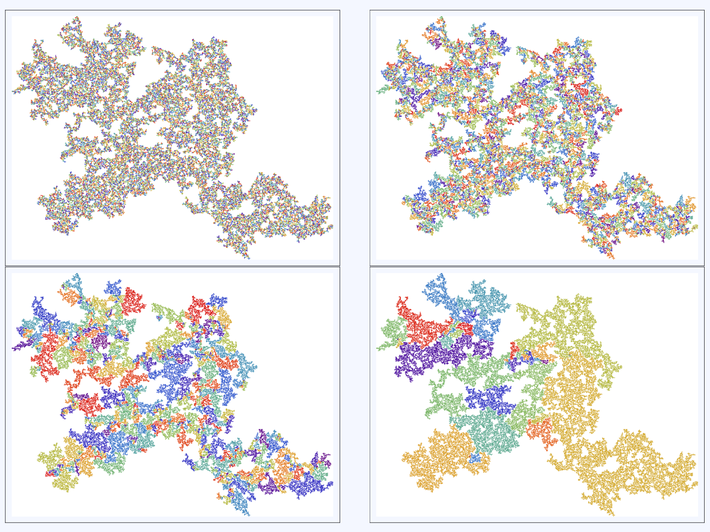


- The true numbers of paths are calculate after the counting by correctly combining the paths through the children with those actually counted

Hierarchical decomposition

After the decomposition the pieces must fulfill two requirements in order to keep storage and CPU time low.

1. The number of links to children and parent must be small.
2. The number of "raw" sites outside the children must be small.

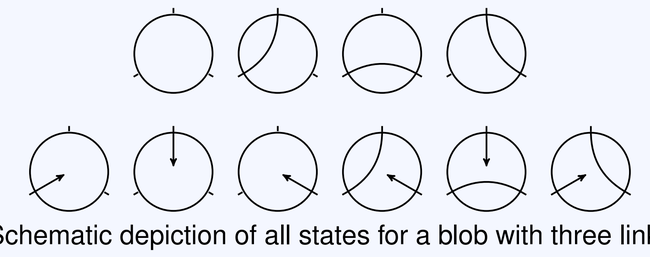


Hierarchy of the pieces is created bottom-up:

1. Define smallest blobs
2. Judge all options ("moves") to merge neighboring blobs
3. Perform "best" move (if possible)
4. Judge new moves, go back to 3.

Topologically different paths

When a blob is enumerated, topologically different ways to transverse the blob and its children need to be kept in separate classes ("states").



Schematic depiction of all states for a blob with three links

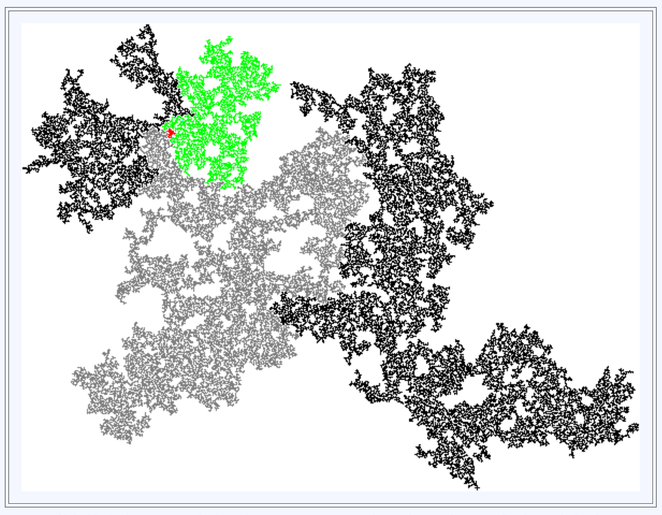
The number of possible states S increases strongly with the number of links L:

$$S = \sum_{i=0}^{L/2} \binom{2i}{i} \binom{L}{2i} (L-2i+1)/(i+1)$$

For $L = 15$, S already exceeds two million. This drastic increase sets a practical limit on the affordable number of links.

Scale-free Enumeration

Central Problem: How to count the number of paths for each topologically distinct way to traverse blob containing already enumerated pieces?



Child (green) behaves as single site of the blob (red and green area)

- Start enumerating from each link to the blob's parent in turn.
- Treat children and parent as "special" sites, that change state when visited.
- Store numbers of paths separately for each distinct combination of states.
- Once done, calculate real number of paths for each state of the blob, by fitting the right segments together.

Performance

The number of different paths of length $N = 600$ on the configuration above (start near center) is $5.388.. \cdot 10^{103}$.

- Decomposition and enumeration took about 20 minutes.
- Best MC methods would need days (maybe weeks) to achieve 1% accuracy.
- Normal exact enumeration would take about 10^{90} years.

First results:

Steps N	Sample size	$\overline{time}$ $[s]$	$\langle R^2 \rangle$	$\langle log_{10}(Z) \rangle$
50	2000	0.8	$3.57(3) \cdot 10^2$	9.06(3)
100	1500	3.4	$1.05(1) \cdot 10^3$	17.28(4)
200	1000	32	$3.11(10) \cdot 10^3$	33.8(1)
300	500	141	$5.66(13) \cdot 10^3$	50.3(1)
400	500	330	$8.9(2) \cdot 10^3$	66.6(2)
500	280	687	$1.27(4) \cdot 10^4$	82.6(3)
600	43	1260	$1.9(2) \cdot 10^4$	98.1(6)

The method is very young, and thorough tests still need to be carried out.

Outlook

Distributions
One great benefit of exact enumeration over MC is that it can yield not only mean values, but complete distributions. This is vital when dealing with extensions of the SAW that include simple interactions, because it allows exact reweighting over the whole temperature range (see [5]). In theory this should be possible for our method as well, although it will probably require substantially more effort.

Higher dimensions
The current implementation is only for two dimensions but can probably be adapted easily. In theory all the ideas apply for higher dimensions too, but whether the method will work in practice still remains to be seen.

References

[1] K. Barat and B. Chakrabarti, *Phys. Rep.* 258 377-411 (1995)
[2] B. Chakrabarti and J. Kertesz, *Phys. Rev. B* 79 221 (1981)
[3] K. Kremer, *Z. Phys. B.* 45, 149-152 (1981)
Z. Phys. B 55, 41-44 (1984)
[4] V. Blavatska and W. Janke, *EPL* 82, 66006 (2008)
[5] S. Kumar et. al., *Phys. Rev. Lett.* 98, 128101 (2007)
[6] P. M. Lam, *J. Phys. A* 23, L831-L836 (1990)
[7] S. B. Lee and H. Nakanishi, *J. Phys. A 24*, 1355-1361 (1991)

Gedruckt im Universitätsrechenzentrum Leipzig