SLAC – PUB – 4075
September 1986
(T)

PERCOLATION AND THE COMPLEXITY OF GAMES*

CONSTANTIN P. BACHAS[†]

Stanford Linear Accelerator Center
Stanford University, Stanford, California, 94305

and

WILFRIED F. WOLFF[‡]

Department of Applied Physics
Stanford University, Stanford, California 94305

ABSTRACT

The complexity of games and certain search algorithms is analyzed and shown to be a measure of the absence of self-similarity of the underlying tree. We establish the connection to the problem of percolation on arbitrary trees.

Submitted to *Journal of Physics A (Letters)*

[*] Work supported in part by the Department of Energy, contract DE-AC03-76SF00515, and by the NSF-MRL Program to the Center for Materials Research at Stanford University.

[†] Present address: Centre de Physique Théorique, Ecole Polytechnique, 91128 Palaiseau, France.

[‡] Address after October 1: IFF/KFA, Postfach 1913, 5170 Jülich, Federal Republic of Germany.

A large number of natural and artificial systems have an exact or approximate hierarchical organization. A useful tool in analyzing them has been the renormalization group and the concept of self-similarity. It is, however, precisely the absence of self-similarity which characterizes the most complex and interesting systems. In an effort to classify hierarchical structures Huberman and Hogg (Huberman and Hogg 1986) introduced a notion of complexity which measures the degree of non-isomorphism at all levels of the hierarchy and which has the property of being minimized by both regular (= uniform) and totally random trees. A physical manifestation of this notion of complexity is the speed of relaxation of hierarchical structures (Bachas and Huberman 1986): self similar ones relax fastest and structural noise is irrelevant. Similar ideas in a different context, that of forecasting the itinerary sequence in a chaotic system, have also been advocated for by Grassberger (Grassberger 1986).

In this letter we will consider yet another manifestation of this notion of complexity, namely in the context of percolation on arbitrary trees and show how it is related to the complexity of games and the time requirement of search algorithms.

Consider a game whose duration is for simplicity taken to be exactly n moves. At each move the player is faced with several options some of which lead to a forced loss assuming an infinitely intelligent opponent. The remaining non-losing strategies form a "winning" tree with n levels of hierarchy. In what follows by tree we shall always mean this "winning" tree. Regular trees correspond to simple games: an example would be a game in which the player is at each step presented with a choice of four colors (red, green, blue and white) and has to pick either blue or white to avoid losing immediately. This is to be contrasted with complex games like, e.g. chess with a fixed number of total moves, for which the tree of non-losing strategies is certainly not regular since certain bad moves might e.g. allow your opponent to checkmate you for the rest of the game thereby limiting your options considerably. In this particular case the lack of self-similarity of the tree is a manifestation of the complexity and history dependence of the game:

put differently, the checkmate positions commonly found in newspapers are not equally hard.

Let us now give a precise and intuitive definition of the complexity of a game. Consider a monkey which at each step of the game decides at random to pursue every one of the available options with probability p. The monkey does not lose if throughout the game it has pursued at least one of the non-losing strategies. When the number of moves becomes arbitrarily large this can only happen if p is greater than the threshold p_c for percolation from the root of the tree to a leaf at its bottom. The smaller p_c, the simpler the game, since the randomly playing monkey may pursue fewer options and still not lose. We may thus define p_c as a measure for the complexity of the game, a definition that we will further qualify later on.

To study percolation on general trees we will use the fact that the probability $Q(t)$ that no path percolates from the root to a leaf of a subtree t satisfies the recursion relation

$$Q(t) = (1 - p + p \cdot Q(t_1)) \ \ldots \ (1 - p + p \cdot Q(t_b)) \tag{1}$$

where b is the number of branches emanating from the root of t and $t_1, \ldots, t_b$ label the corresponding subtrees (see Fig. 1).

We first consider the simplest case, namely percolation on regular or uniform trees, which have the same branching ratio b at each node of every generation. Variations of this problem have already been considered to study the mean-field theory of percolation (Essam-Fisher 1961, Essam 1980, Stauffer 1979), more recently in the context of spreading activation in computer networks (Hogg and Huberman 1986) and also to study the time requirements of certain heuristic searches (Karp and Pearl 1983, Stone and Sipala 1986). The probability Q_n that in a tree with n generations no path percolates then satisfies the recursion relation

$$Q_n = (1 - p + p \cdot Q_{n-1})^b \quad . \tag{2}$$

Equation (2) has the fixed point $Q = 1$, corresponding to no percolation. This fixed point becomes unstable for $p > p_c^{\text{uniform}} = 1/b$, signalling the appearance with a finite probability of a percolating cluster from the root to the bottom of the tree.

Clearly, the fatter the tree, i.e. the larger the branching ratio b, the smaller the percolation threshold, since fat trees are harder to cut. Although one could argue that the complexity of a game should increase whenever some winning strategies are discarded (e.g. keeping only white as a winning choice in the aforementioned game) this is a rather trivial effect. Therefore we will in the sequel limit ourselves to games whose trees have a fixed average branching ratio b per generation or a fixed multiplication rate of non-losing strategies per move. This is reminiscent of ultradiffusion where fatter trees relax faster and where one must fix the tree silhouette in order to study the effect of the tree structure on the dynamics (Bachas and Huberman 1986).

We shall now show that p_c is indeed minimized for regular, i.e. uniform trees as it should be since these trees correspond to the simplest games.

Theorem: The critical threshold for percolation p_c in a tree with average branching ratio b satisfies

$$p_c \geq p_c^{\text{uniform}} = \frac{1}{b} \quad .$$

Proof: The probability for no percolation on a generic tree with average branching ratio b satisfies

$$Q = (1 - p + p \cdot Q(t_1)) \ldots (1 - p + p \cdot Q(t_b)) \geq \left(1 - p + p \cdot \sqrt[b]{Q(t_1) \ldots Q(t_b)}\right)^b$$

where $t_1, \ldots, t_b$ are the subtrees with roots at the first hierarchy level. The inequality follows from the well known fact that the arithmetic mean is always bigger or equal to the geometric mean. At the second step we have

$$Q(t_1)\dots Q(t_b) \;\geq\; \left(1-p+p\cdot\sqrt[b^2]{Q(\tilde{t}_1)\dots Q(\tilde{t}_{b^2})}\right)^{b^2}$$

where $\tilde{t}_1,\dots,\tilde{t}_{b^2}$ are the subtrees with roots at the second hierarchy level; this follows from the recursion relation (1) and the above inequality.

Repeating this procedure n times we finally get

$$Q \;\geq\; f\circ\dots\circ f(0) \qquad (n \text{ times})$$

with $f(x) = (1-p+px)^b$. The right-hand side is the probability for no percolation on a uniform tree, which completes the proof.

We now consider the case of random trees which are constructed by allowing the branching ratio x at every node to be an independent random variable with distribution $P(x)$ and average value $\langle x\rangle = b$. After deleting bonds with probability $1-p$ the cluster connected to the root is another random tree with modified probability distribution

$$\widetilde{P}(y) = \sum_{y}^{\infty} P(x)\cdot\begin{pmatrix} x \\ y \end{pmatrix}\; p^y(1-p)^{x-y}$$

and average branching ratio $\langle y\rangle = p\cdot b$; here $\binom{x}{y}$ is the binomial coefficient. The probability for percolation on the original tree is equal to the probability that the modified random tree survives for an infinite number of generations. From the theory of branching processes (Harris 1963) it is known that the latter probability is finite when the average branching ratio is bigger than one and zero when it is smaller than one, from which we deduce that the critical probability for percolation on the original tree is given by $\langle y\rangle = p_c\cdot b = 1$.

The above argument shows that random trees with average branching ratio $\langle x\rangle = b$ have the same percolation threshold as uniform trees with branching ratio b. We have thus demonstrated that the complexity of a game which was defined as the percolation threshold for the underlying tree is sensitive to the absence of self-similarity rather than to the existence of noise. This is in qualitative agreement with the measure of complexity introduced by Huberman and Hogg (Huberman and Hogg 1986) as well as with the dynamic exponent that characterizes ultradiffusion with long range hoppings (Bachas and Huberman 1986).

In order to show that the absence of self-similarity does indeed lead to a larger value for p_c and thus to an enhanced complexity we shall now consider as a specific example the highly unbalanced tree shown in Fig. 2. At each generation the left half nodes give rise to $2b - 1$ offsprings, while the rest continue as dead branches to the bottom so that the average branching ratio is b. We will now show that the percolation threshold is equal to one for this particular tree which thus describes the winning strategies of a game of maximal complexity. The argument runs as follows: if $p < 1$ then the leftmost path from the root to the bottom is cut with probability one at a finite hierarchy level n away from the root; the remaining $b^n - 1$ nodes generate subtrees which $(2b - 1)$-furcate for a finite number of steps and then continue as dead branches thereafter. Clearly these subtrees cannot percolate if $p < 1$ so that $p_c = 1$ as claimed above.

An intuitive explanation why this game is of maximal complexity is that there is zero tolerance for errors since the player must either follow the strategy corresponding to the leftmost branch or else will eventually be lead to a situation where non-losing moves are unique. This same class of trees also leads to slowest relaxation in ultradiffusion (Bachas and Huberman 1986); it would be interesting to see if there exists a quantitative relationship between the dynamical critical exponent and the percolation threshold p_c, i.e. the complexity of games.

Finally we would like to point out another possible interpretation of our results, namely in the context of heuristic searches. Following Karp and Pearl (Karp and Pearl 1983) one can consider the problem of finding an optimal path on a tree whose edges are assigned the cost 1 with probability p and 0 with probability $1-p$. A 'uniform cost' algorithm that performs this task is one that starting from its leftmost node expands the front of nodes reachable with a given cost from the root of the tree until it finds a leaf at the bottom of the tree. Karp *and Pearl show that for regular trees this algorithm runs efficiently (linear time* requirement) if the 0-cost edges have a finite probability to percolate and has an exponential time requirement otherwise. Since, as shown above, these 0-cost edges can never percolate on a sufficiently complex tree we would then expect the running requirement for these trees to be always exponential. It would be very interesting to analyze this problem in more detail but this is beyond the scope of this paper.

We thank Bernardo Huberman for many enlightening discussions.

References

Bachas, C.P. and Huberman, B.A., 1986, SLAC-PUB-4016 (June 1986); preprint (September 1986).

Essam, J.W., 1980, Rep. Prog. Phys. 43, 833.

Fisher, M.E. and Essam, J.W., 1961, J. Math. Phys. 2, 609.

Grassberger, P., 1986, talk at 16th Stat. Physics Conference, Boston University, August 9-16.

Harris, T., 1963, *The Theory of Branching Processes* (Springer, Berlin).

Hogg, T.H. and Huberman, B.A., 1986, Xerox Palo Alto preprint.

Huberman, B.A. and Hogg, T.H., 1986, Physica D (August 1986).

Karp, R.M. and Pearl, J., 1983, Artif. Intell. 21, 99.

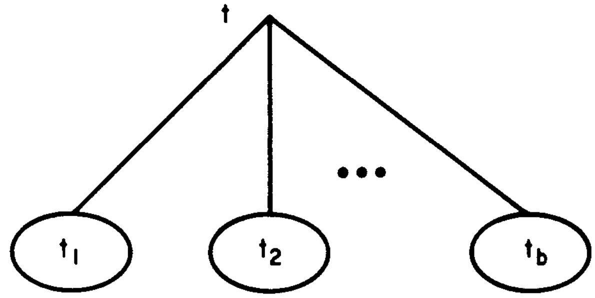
Stauffer, D., 1979, Phys. Rep. 54, 1.

Stone, H.S. and Sipala, P., 1986, IBM J. Res. Develop. 30, 242.

Figure Captions

Fig. 1. A tree t and its subtrees $t_1, \ldots, t_b$ at the first hierarchy level, used in the recurrence relation (1).

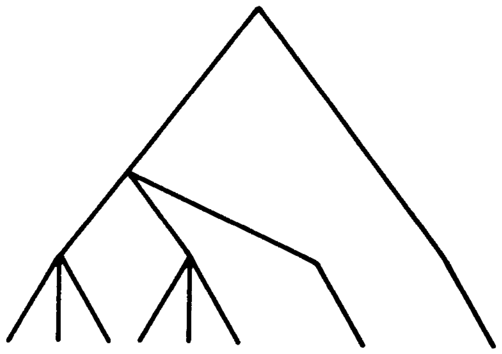
Fig. 2. An example of a maximally complex tree; the left-half nodes of every generation give rise to 3 offsprings, while the rest continue as dead branches to the bottom. The average branching ratio is $b = 2$.



9-86

5539A1

Fig. 1



9-86

5539A2

Fig. 2