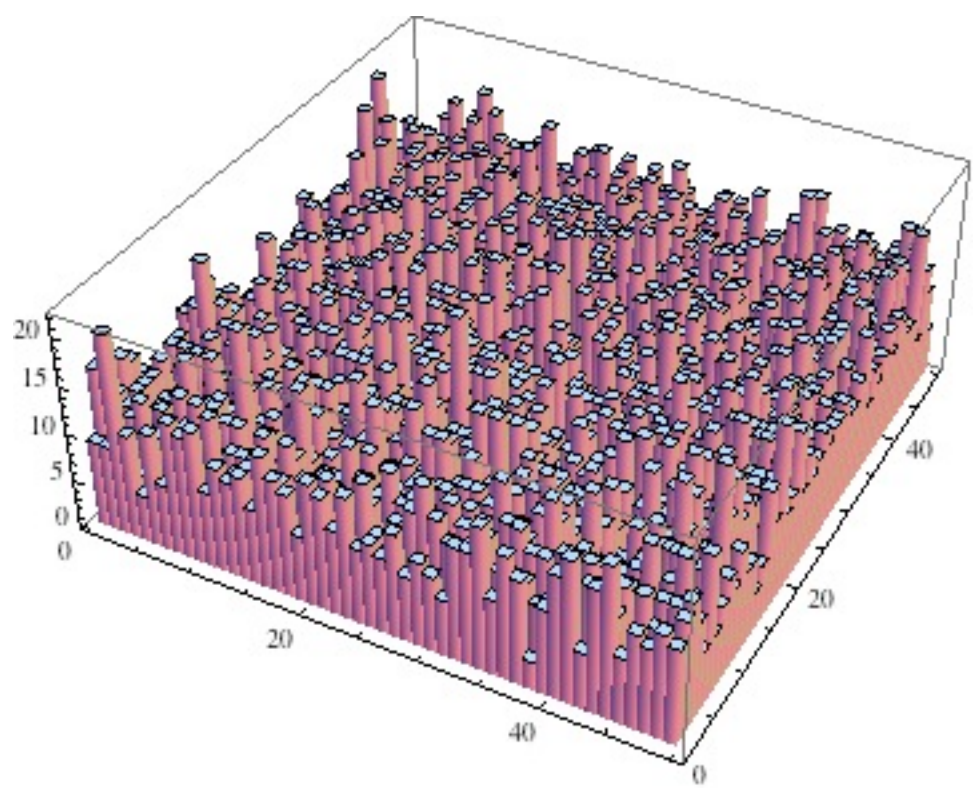
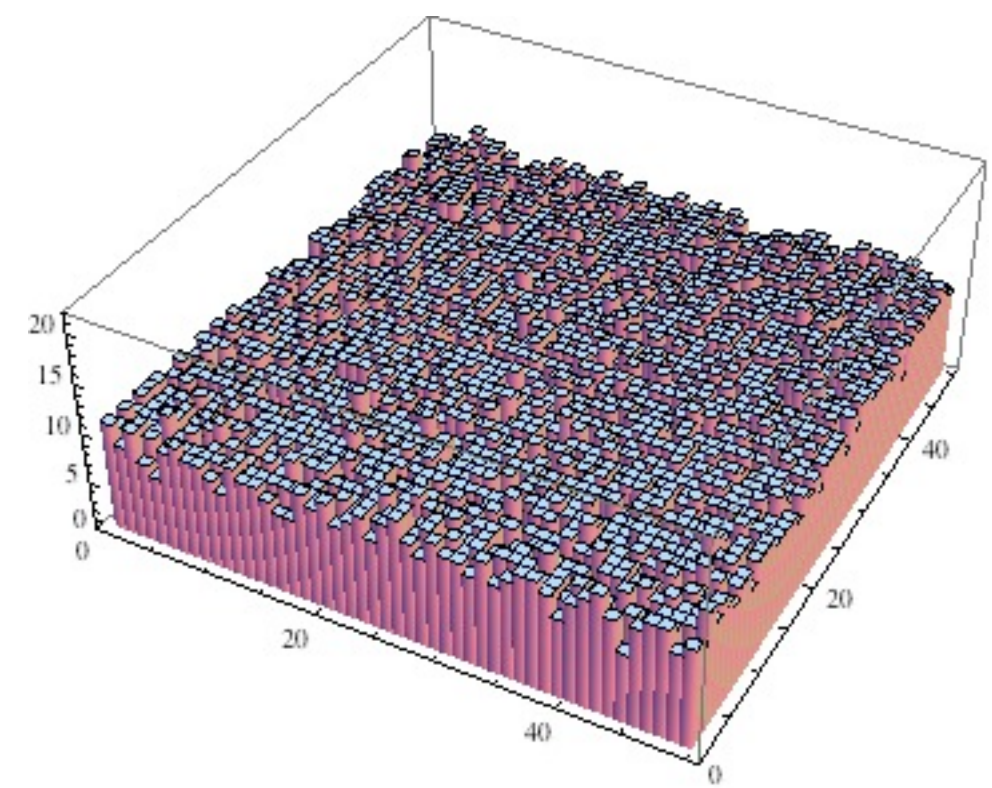
The Power of Choice in Random Satisfiability

Varsha Dani, University of New Mexico
Josep Díaz, Universitat Politècnica de Catalunya
Thomas Hayes, University of New Mexico
Cristopher Moore, Santa Fe Institute

25,000 balls in 2,500 bins



uniformly random:
O(log n) fluctuations



smaller of two bins:
O(log log n) fluctuations

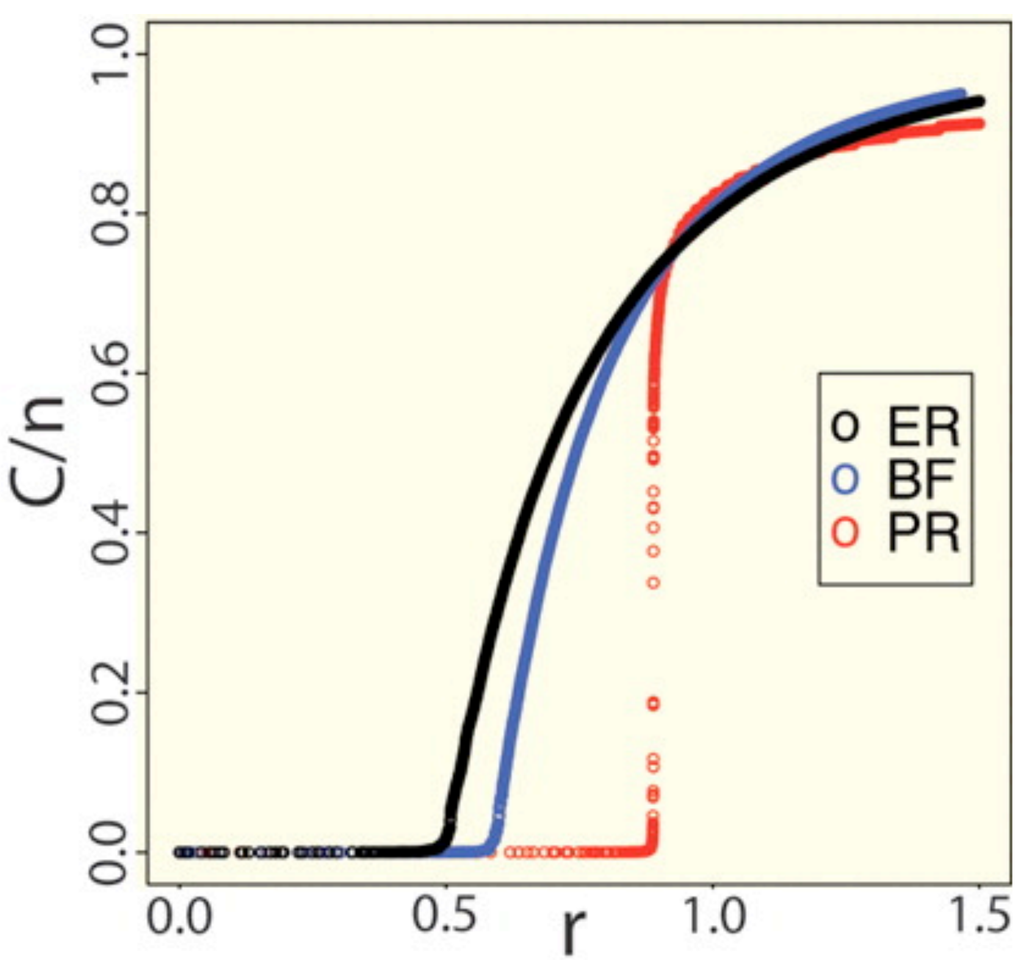
[Azar, Broder, Karlin, Upfal; Mitzenmacher]

Explosive percolation

Given two uniformly random edges, choose one (online) to add to the graph

Goal: delay the emergence of the giant component

One strategy: join the pair of components with the smaller product of their sizes



[Bohman & Frieze;
Spencer & Wormald;
Achlioptas, D'Souza, Spencer;
Riordan & Warnke]

The phase transition for k-SAT

$F_k(n,m)$: a k-SAT formula with n variables and m clauses, chosen independently and uniformly from the $2^k n^k$ possible clauses

Threshold conjecture: for every $k \geq 3$, there is a constant α_k such that

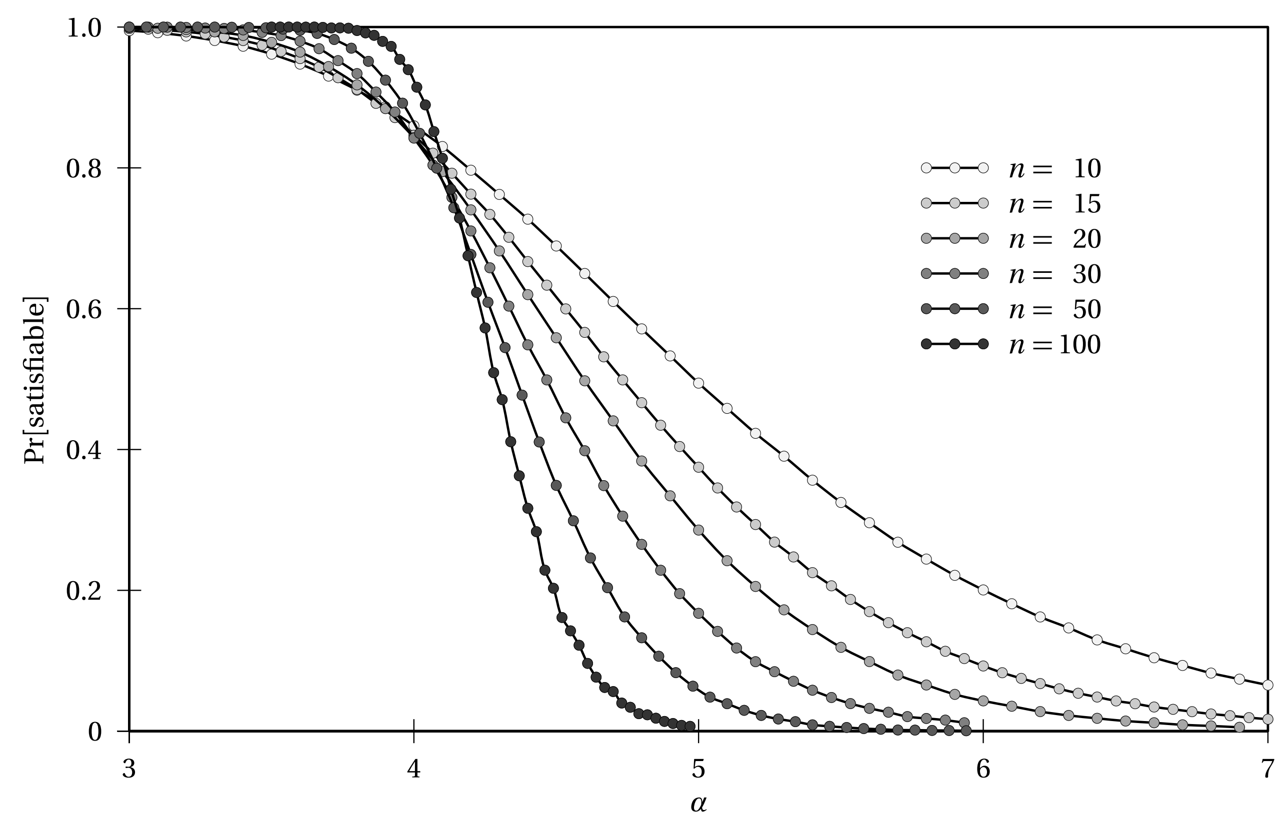
$$\lim_{n\to\infty} \Pr[F_k(n, m = \alpha n) \text{ is satisfiable}] = \begin{cases} 1 & \text{if } \alpha < (1-\epsilon)\alpha_k \\ 0 & \text{if } \alpha > (1+\epsilon)\alpha_k \end{cases}$$

Nonuniform threshold $\alpha_k(n)$ [Friedgut]

$3.52 \leq \alpha_3 \leq 4.4898$ [KKL, HS, DKMP]

First-moment upper bound: $\alpha_k < 2^k \ln 2$

For large k, $\alpha_k = 2^k \ln 2 - O(1)$
[Achlioptas & Moore, Achlioptas & Peres, Coja-Oghlan & Panagiotou]



Achlioptas processes for *k*-SAT

Choose each clause online between *h* uniformly random *k*-SAT clauses

Can we move the threshold $\alpha=m/n$ at which the formula becomes unsatisfiable?

[Sinclair & Vilenchik] 2 choices can raise the threshold for $k=2$ and $k=\omega(\log n)$

[Perkins] for each *k*, there is an *h* such that *h* choices raise the *k*-SAT threshold: $h=7$ suffices for all k, and $h=3$ suffices for sufficiently large *k*

Our contributions:

- 3 choices suffice to raise the *k*-SAT threshold for all *k*
- 2 choices suffice for $3 \le k \le 50$
- 2 choices suffice to lower the threshold, if there is one...

Positive thinking

Our rule: given h clauses, choose the one with the most positive literals

Simple, nonadaptive, and oblivious to topology: doesn't depend on which variables have appeared before, or with what values

Denote the resulting formula $F_{k,h}$

[Perkins] To prove that $F_{k,h}$ is satisfiable: form a 2-SAT formula, taking two literals from each k-clause in $F_{k,h}$

If this formula is satisfiable, then $F_{k,h}$ is too

Our rule: take two of the most positive literals from each k-clause

Positive thinking, continued

c=the most positive 2-SAT clause in the most positive of *h* *k*-SAT clauses

Probability that *c* has 0, 1, or 2 positive literals:

$$p_0 = 2^{-kh}$$
$$p_1 = \left(2^{-k}(k+1)\right)^h - p_0$$
$$p_2 = 1 - p_0 - p_1\,.$$

What is the threshold for biased 2-SAT formulas with *m*=*αn* clauses?

Branching process of unit clauses, e.g. $x \wedge (\neg x \vee y) \Rightarrow y$: $\alpha \begin{pmatrix} p_1 & 2p_0 \\ 2p_2 & p_1 \end{pmatrix}$

Threshold occurs when largest eigenvalue is 1 [Mossel, Sen]: $\alpha = \dfrac{1}{p_1 + 2\sqrt{p_0 p_2}}$

Three choices suffice

In our case, this gives a lower bound that grows as $\alpha \approx 2^{kh/2}$

For $k \geq 4$, this exceeds the first-moment upper bound $\alpha_k \leq 2^k \ln 2$
(for k=3 we need improved upper bounds)

So h=3 choices are enough to raise the threshold for all k

What about h=2?

2-SAT subclauses aren't powerful enough...

Using 3-SAT clauses instead

Form a 3-SAT formula by taking three of the most positive literals from each k-clause in $F_{k,h}$

Biased random formula: $\alpha n p_j$ clauses with j positive literals for each j=0, 1, 2, 3

$$p_0 = 2^{-kh}$$
$$p_1 = \left(2^{-k}(k+1)\right)^h - p_0$$
$$p_2 = \left(2^{-k}\left(\binom{k}{2} + k + 1\right)\right)^h - p_1 - p_0$$
$$p_3 = 1 - p_0 - p_1 - p_2\,.$$

If the resulting 3-SAT formula is satisfiable, then so is $F_{k,h}$

A Biased Unit Clause algorithm

Set variables permanently, one at a time: no backtracking

(Forced step) If there are any unit clauses, choose one uniformly and satisfy it

(Free step) Else, choose x uniformly from the unset variables, and set x=**true**

BUC fails if a contradictory pair of unit clauses appears

We will use differential equations to show that, with constant probability, this doesn't occur: high probability then follows from Friedgut

Setting x=true removes c if $x \in c$, and shortens c if $-x \in c$

During the algorithm, we have a mix of 3-, 2-, and unit clauses: how many?

Differential equations

$S_{i,j}(T)$=number of i-clauses with j positive literals after T variables have been set

$q_0(T)$, $q_1(T)$=probability that the variable on step T is set false or true

$$\mathbb{E}[\Delta S_{3,j}] = -\frac{3S_{3,j}}{n-T}$$
$$\mathbb{E}[\Delta S_{2,j}] = \frac{(3-j)q_1 S_{3,j} + (j+1)q_0 S_{3,j+1} - 2S_{2,j}}{n-T}$$

Rescale to $s_{i,j} = S_{i,j}/n$, $t=T/n$:

$$\frac{\mathrm{d}s_{3,j}}{\mathrm{d}t} = -\frac{3s_{3,j}}{1-t}$$
$$\frac{\mathrm{d}s_{2,j}}{\mathrm{d}t} = \frac{(3-j)q_1 s_{3,j} + (j+1)q_0 s_{3,j+1} - 2s_{2,j}}{1-t}$$

[Wormald] w.h.p. $S_{i,j}(T) = s_{i,j}(t)n + o(n)$

A branching process of forced steps

Branching process of unit clauses: $M = \frac{1}{1-t} \begin{pmatrix} s_{2,1} & 2s_{2,0} \\ 2s_{2,2} & s_{2,1} \end{pmatrix}$

We succeed with positive probability iff M's largest eigenvalue $\lambda < 1$ for all t

Cascade of forced steps, starting with a free step

Total expected number of variables set false or true:

$$\begin{pmatrix} b_0 \\ b_1 \end{pmatrix} = \left(\mathbb{1} + M + M^2 + \cdots\right) \cdot \begin{pmatrix} 0 \\ 1 \end{pmatrix} = (\mathbb{1} - M)^{-1} \cdot \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

Probability a variable is set false or true on a given step:

$$q_0 = \frac{b_0}{b_0 + b_1}, \; q_1 = \frac{b_1}{b_0 + b_1}$$

Two choices suffice for $5 \le k \le 50$

We integrate these differential equations numerically, with initial conditions

$$s_{3,j}(0) = \alpha p_j \,,\; s_{2,j}(0) = 0$$

For each k, we find the largest α such that $\lambda < 1$ for all t

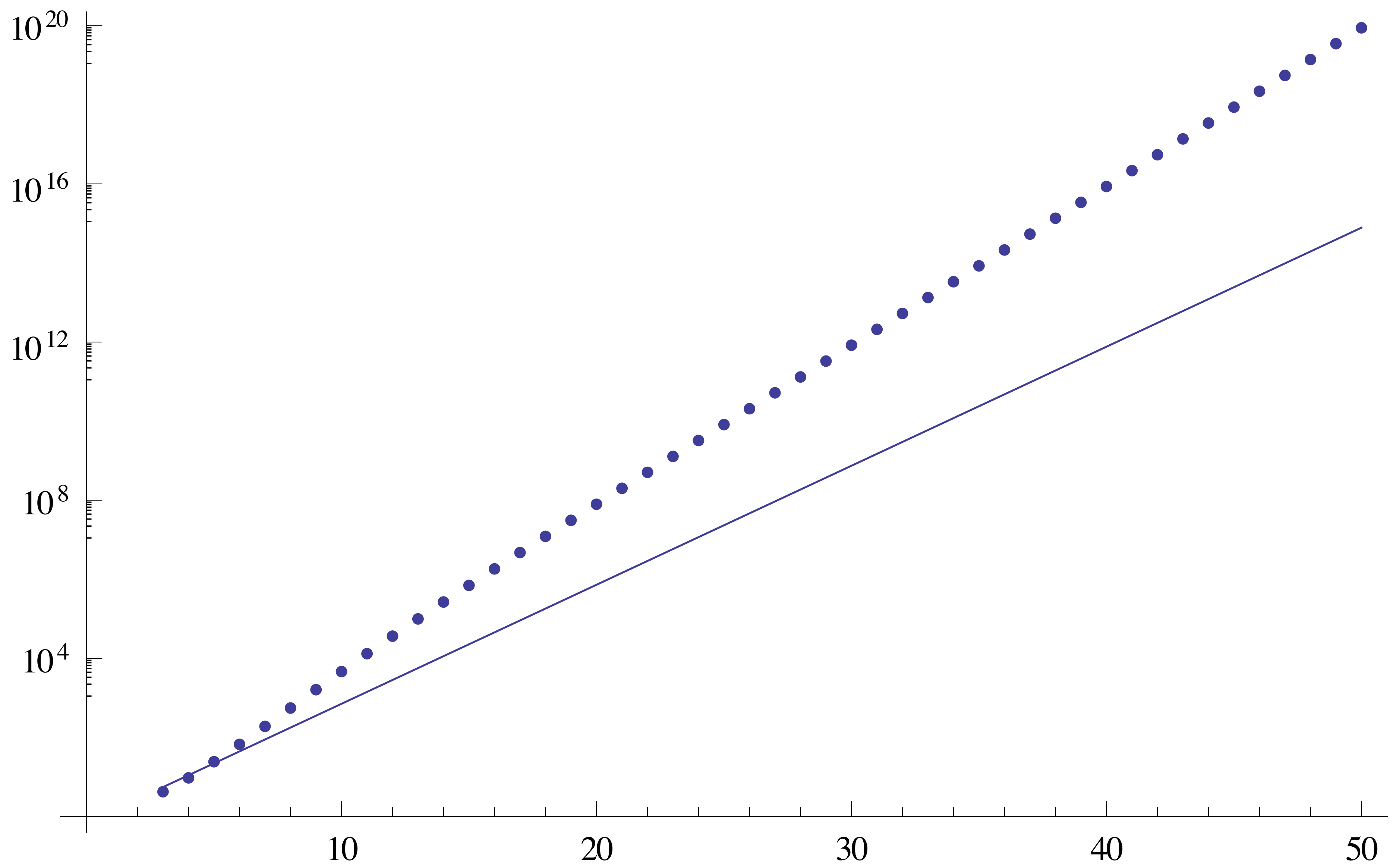
Regrettably, we have to do this calculation separately for each k...

For $5 \le k \le 50$, this lower bound α_{BUC} exceeds first-moment upper bound on α_k

k	3	4	5	6	7	8	9	10
α_{BUC}	4.232	9.491	24.306	66.811	190.806	554.106	1610.88	4637.05
$2^k \ln 2$			22.18	44.36	88.72	177.45	354.89	709.78

For $k > 50$, we need something more analytic...

Conjecture: two choices suffice for $k > 50$ too



α_{BUC} vs. $2^k \ln 2$ for $3 \leq k \leq 50$

Two choices suffice for k=4

For k=3 and k=4, we need better algorithms

For k=4, don't extract 3-clauses: run BUC directly on biased 4-SAT formulas

Differential equations for $s_{4,j}$, $s_{3,j}$, and $s_{2,j}$

Same branching process for unit clauses

We find $\lambda < 1$ for all t for α=10.709...

...which exceeds an upper bound $\alpha_4 \leq 10.217$ based on counting locally maximal assignments [Dubois & Boufkhad]

Two choices suffice for k=3 (finally!)

Biased Short Clause algorithm:

(Forced step) If there exist unit clauses, choose one uniformly and satisfy it

(Free step) Else, if there are any 2-clauses, choose one uniformly

- If it has any positive literals, choose one uniformly and satisfy it
- Else, if both its literals are negative, choose one uniformly and satisfy it

(Really free step) If there are no unit clauses or 2-clauses, choose x uniformly from the unset variables and set x uniformly

Fancier differential equations: p_{free} depends on expected length of cascade

We find $\lambda < 1$ for all t for α=4.581...

...but best upper bound is $\alpha_3 \leq 4.4898$ [Díaz, Kirousis, Mitsche, Pérez-Giménez]

The story so far

We have shown that a simple strategy raises the k-SAT threshold for all k if we have 3 choices, and for $k \leq 50$ if we have 2 choices

Moreover, our proof shows that these formulas are *easy to satisfy* at densities above α_k: linear-time algorithms (2-SAT or greedy algorithms like BUC and BSC)

Strong numerical evidence that 2 choices suffice for all k

We end with a simple way to lower the threshold, if there is one...

Lowering the threshold

Make denser formulas

Choose constant $b < 1$, and prefer clauses with just the first bn variables

With h choices, we get such a clause with probability $1-(1-b^k)^h$

Ignore other clauses! A subformula on $n'=bn$ vars with expected density

$$\alpha' = \frac{m'}{n'} = \frac{1-(1-b^k)^h}{b}\alpha$$

With $h=2$, we have $\alpha' > \alpha$ if we set $b=((2k-2)/(2k-1))^{1/k}$

This subformula becomes unsatisfiable when $\alpha'=\alpha_k$, but $\alpha < \alpha_k$

Also a simple strategy for speeding up the birth of the giant component (not as good as Spencer & Wormald)

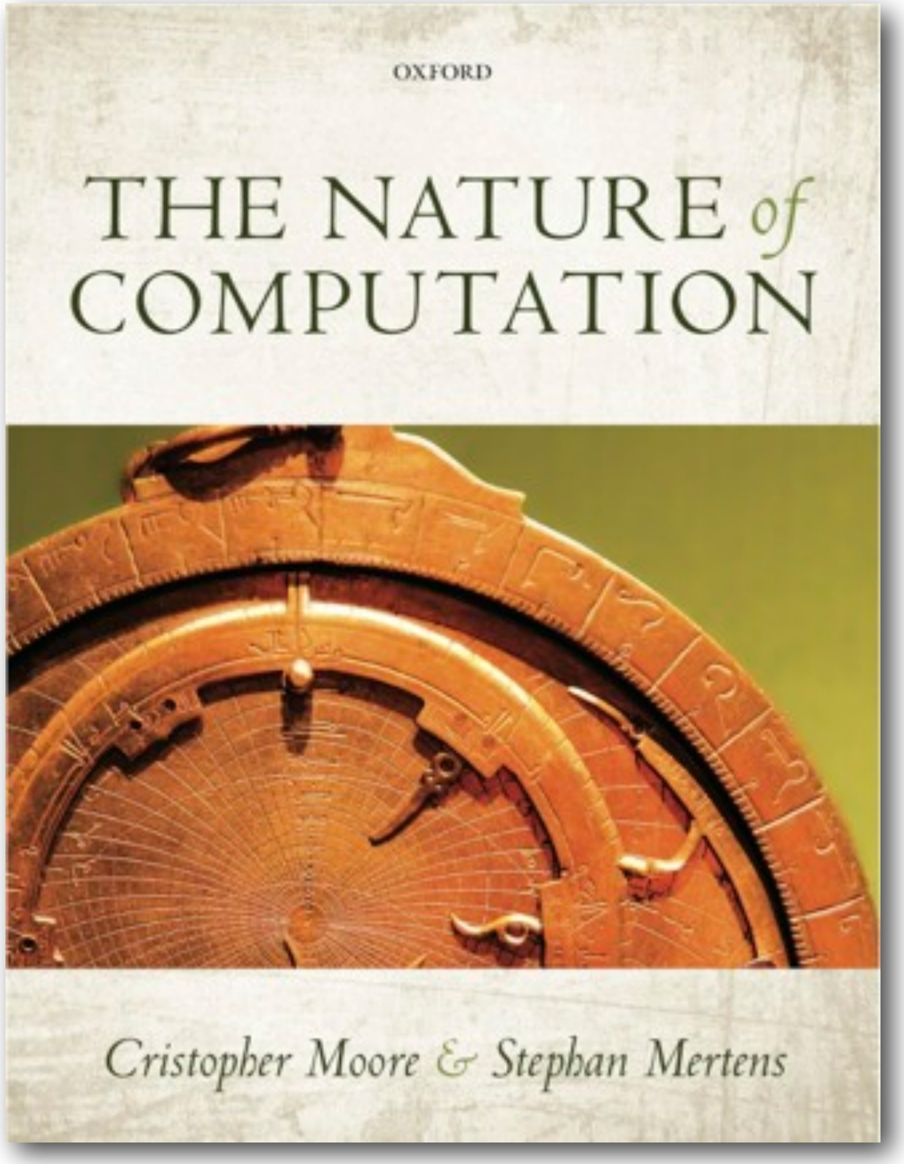
Open questions

Do two choices suffice to raise the k-SAT threshold for all k?

Do two choices suffice to lower the k-SAT threshold, if we don't assume the threshold conjecture?

[Achlioptas] Are there any interesting graph-theoretic properties for which no bounded-size strategy with two choices changes the threshold?

Shameless Plug



www.nature-of-computation.org

To put it bluntly: this book rocks! It somehow manages to combine the fun of a popular book with the intellectual heft of a textbook.

Scott Aaronson, MIT

A creative, insightful, and accessible introduction to the theory of computing, written with a keen eye toward the frontiers of the field and a vivid enthusiasm for the subject matter.

Jon Kleinberg, Cornell

A treasure trove of ideas, concepts and information on algorithms and complexity theory. Serious material presented in the most delightful manner!

Vijay Vazirani, Georgia Tech

A fantastic and unique book, a must-have guide to the theory of computation, for physicists and everyone else.

Riccardo Zecchina, Politecnico de Torino

This is the best-written book on the theory of computation I have ever read; and one of the best-written mathematical books I have ever read, period.

Cosma Shalizi, Carnegie Mellon

Acknowledgments

and NSF, DARPA/AFOSR, ARO, and NIST