The Exact Satisfiability Threshold for a Potentially Intractable Random Constraint Satisfaction Problem

Harold Connamacher
University of Toronto
Department of Computer Science
hsc@cs.toronto.edu

Michael Molloy*
University of Toronto
Department of Computer Science
molloy@cs.toronto.edu

Abstract

We determine the exact threshold of satisfiability for random instances of a particular NP-hard constraint satisfaction problem. The problem appears to share many of the threshold characteristics of random k-SAT for $k \geq 3$; for example, we prove the problem almost surely has high resolution complexity. We also prove the analogue of the $(2+p)$-SAT conjecture for a class of problems that includes this problem and XOR-SAT.

1. Introduction

Determining the satisfiability threshold for random k-SAT is a fundamental problem that has received attention from several scientific communities. (See, eg. [1] for a survey of the area.) The basic question is this: determine a constant c_k such that a uniformly random instance of k-SAT with n variables and cn clauses will be almost surely (a.s.) satisfiable if $c < c_k$ and a.s. unsatisfiable if $c > c_k$. A property holds *almost surely* (a.s.) if its probability tends to 1 as the number of variables tends to ∞. The case $k = 2$ is well-understood - $c_2 = 1$[11, 28, 21]. But for $k \geq 3$ it is not even known whether c_k exists, although Friedgut[23] proved something close. We know that for $c < 3.52$[29, 31] the formula is a.s. satisfiable, while for $c > 4.506$[18] it is a.s. unsatisfiable, so if c_k exists it satisfies $3.52 \leq c_k \leq 4.506$.

This is a mathematically beautiful problem, and as a result, it has attracted much attention from mathematicians and theoretical computer scientists. Furthermore, it is also widely studied by researchers, mostly in AI, who build SAT-solving tools, and in statistical physics, who model thresholds from nature. For AI researchers, it has long been observed[10, 38] that this random model provides a rich source of difficult problems if the number of clauses is close to $c_k n$ (or more correctly, to an experimentally determined conjecture for the value of $c_k n$). For physicists (see eg. [34, 35, 46, 45]), there is a correspondence between problems such as random k-SAT and models used to study threshold behaviors of natural processes, such as that of water when its temperature passes through the "threshold of freezing". So they regard random k-SAT as an important mathematical object, the understanding of which will provide insights into physical phenomena.

This interest spread to generalizations of k-SAT. For example, the Schaefer[47] generalizations inspired studies of random instances of problems such as 1-in-k-SAT[3], NAE-k-SAT[3, 6], k-XOR-SAT[19, 36]. Generalizing further to allow the variables to take values from a domain of size larger than 2 led to the study of various models for random constraint satisfaction problems[5, 16, 37, 39, 40]. In all of these cases, the primary focus has been to try to determine the satisfiability threshold. For empirical results, there is a large body of work by researchers building CSP-solving tools to experimentally approximate the satisfiability thresholds of various models of random CSPs and study the difficulty of random instances with constraint density close to that threshold. (See [27] for a survey of several such studies; see also [9] for some studies from the statistical physics community.) The reasons for the study of these generalizations include: (1) the generalizations are interesting and beautiful problems in their own right, and (2) the study of these generalizations can often yield insights into random 3-SAT. (See [6] for a very fruitful instance of (2)).

Thresholds for models where the domain size grows with n[20, 25, 48, 49, 50] and where the constraint size grows with n[22, 24] have been studied. Such models seem to, at heart, be substantially different than random 3-SAT. For one thing, the satisfiability thresholds occur when the number of clauses/constraints is superlinear in the number of variables, and this produces a structure that is very different than that of random 3-SAT when the number of clauses is near the satisfiability threshold and, hence, is linear. Models where

* Much of this work was done while the second author was a guest at the Fields Institute, and a visiting researcher at Microsoft Research.

the domain and constraint sizes are both constant seem to be much closer to random 3-SAT; for example, in [39] it is shown that the satisfiability thresholds of such models occur at a linear number of clauses (see also [16] and [40]).

It is often speculated (see eg. [46]) that the nature of random 3-SAT near its satisfiability threshold may be related to the fact that it is NP-complete. For this reason, along with the fact that many look to these random problems for difficult instances, there is particular interest in determining the satisfiability thresholds for NP-complete problems with constant sized domain and constraints. The threshold is known for a few such problems, eg. 1-in-k-SAT[3], $(2+p)$-SAT for $p < 2/5$[45] and Theorem 5 of [42], but these problems don't shed much light on 3-SAT. In each case, the proof illustrated that these problems are, at least near the satisfiability threshold, very similar to 2-SAT (whose threshold behavior is very different from that of 3-SAT). Furthermore, the proofs involve (at least implicitly) the analysis of fairly simple algorithms. This analysis implies that random instances of the problems are fairly easy to solve, algorithmically.

Dubois and Mandler[19] determined the exact satisfiability threshold for random 3-XOR-SAT, confirming a conjecture that came from the statistical physics community[36]. 3-XOR-SAT is in P as it can be solved by Gaussian elimination modulo 2. Nevertheless, it seems to be much more relevant to random 3-SAT than any of the NP-hard problems from the previous paragraph. Statistical physicists study XOR-SAT (e.g. [13, 36]) because it is exactly the p-spin model, the simplest non-trivial spin-glass like model over random graphs at zero temperature, a family of models which includes random k-SAT[33, 34]. These studies suggest that XOR-SAT has many of the same threshold characteristics of SAT though it is easier to analyze. Furthermore, the proof of the satisfiability threshold for 3-XOR-SAT is non-algorithmic, in that it uses the second moment method and doesn't involve in any way the Gaussian elimination algorithm. It seems to be highly relevant that 3-XOR-SAT does not have a polytime resolution based algorithm (i.e. a DPLL algorithm), even for random instances where the number of clauses is near the satisfiability threshold (see Theorem 2 below). Thus, while 3-XOR-SAT is in P and 1-in-k-SAT is NP-hard, as far as random instances are concerned, 1-in-k-SAT is algorithmically much simpler as it can be solved with a DPLL algorithm.

In this paper, we present an NP-complete constraint satisfaction problem which generalizes XOR-SAT. The problem is called UE-CSP and is defined in the next section. We use d-UE-CSP to refer to UE-CSP with domain size d, and we use (k, d)-UE-CSP to refer to UE-CSP with domain size d and where all constraints have size k. $(k, 2)$-UE-CSP is k-XOR-SAT, and thus is in P; so is $(3, 3)$-UE-CSP. But $(3, 4)$-UE-CSP is NP-complete. We designed this problem so that we could apply the techniques from [19] to prove:

Theorem 1 *The satisfiability threshold for random $(3, 4)$-UE-CSP is $c^* = .917935\ldots$.*

We emphasize that our proof, on both sides of the threshold, is non-algorithmic. Unlike XOR-SAT, we do not know of any efficient algorithm that will a.s. solve instances with clause density slightly less than c^*, nor do we know of any efficient algorithm that will a.s. solve instances with clause density slightly greater than c^*. The following theorem implies that there is no resolution-based algorithm (i.e. DPLL algorithm) of the latter type.

Theorem 2 *For any constant $c > 0$, and any $k \geq 3, d \geq 2$, the resolution complexity of a uniformly random instance of (k, d)-UE-CSP with n variables and cn clauses is a.s. $2^{\Theta(n)}$.*

$(3, 4)$-UE-CSP is the first NP-hard constraint satisfaction problem with constant sized domain and constraints for which (a) Theorem 2 holds and (b) the exact satisfiability threshold is known. Similar results are known for some models whose constraint- or domain-sizes grow with n. In [50] Xu and Li prove that some such models, for whom the satisfiability threshold is known, exhibit high *tree*-resolution complexity. They also note that the model whose satisfiability threshold is determined in [24] is easily shown to exhibit high resolution complexity.

Another important open question concerns $(2+p)$-SAT, a model introduced by [44] with a mixture of clauses of size 2 and 3. The threshold for 2-SAT is 1[11, 28, 21], and in [4], it is proven that a SAT instance with $(1-\epsilon)n$ 2-clauses and αn 3-clauses will be a.s. satisfiable of $\alpha \leq \frac{2}{3}$ and a.s. unsatisfiable if $\alpha \geq 2.28$. It is conjectured that this instance is a.s. unsatisfiable for any $\alpha > \frac{2}{3}$. Besides being a tantalizing conjecture in it's own right, this also has some interesting implications regarding the behavior of many DPLL algorithms on random 3-SAT (see [2]).

We prove the analogue of the $(2+p)$-SAT conjecture for UE-CSP. In [14], the first author proved that the threshold for $(2, d)$-UE-CSP ($d \geq 2$) is $\frac{1}{2}$, and that a d-UE-CSP instance with $\left(\frac{1}{2} - \epsilon\right) n$ 2-clauses and βn 3-clauses will be satisfiable with probability bounded away from zero if $\beta \leq \frac{1}{6}$ and a.s. unsatisfiable if $\beta > \frac{1}{2} + \epsilon$. In this paper, we close the gap.

Theorem 3 *For any $d \geq 2$ and for any constant $\delta > 0$ there exists a constant $\epsilon > 0$ such that a uniformly random instance of d-UE-CSP with $\left(\frac{1}{2} - \epsilon\right) n$ 2-clauses and $\left(\frac{1}{6} + \delta\right) n$ 3-clauses is a.s. unsatisfiable.*

The proof in [4] and in [14] for satisfiability are by showing that the simple greedy algorithm Unit Clause will succeed with probability bounded away from zero. What is conjectured for $(2 + p)$-SAT and now proven for XOR-SAT, and more generally, d-UE-CSP is that, in a sense, Unit Clause is the best we can do on these mixed formulae. That is, more sophisticated algorithms cannot handle any increase in the proportion of 3-clauses if the number of 2-clauses is cn where c is allowed to be arbitrarily close to the threshold for $(2, d)$-UE-CSP, as such a problem will a.s. not even be satisfiable.

In summary, the main contribution of this paper is that we determine the exact satisfiability threshold of a problem that is in many ways similar to random 3-SAT. Furthermore, it is hoped that additional study of this problem, for example a better understanding of how its structure changes near that threshold, will provide some insights into the threshold for random 3-SAT.

2. (k, d)-UE-CSP

2.1. The problem

If we consider XOR-SAT as a constraint satisfaction problem with constraints on k vertices, the key property of XOR-SAT which permits the technique of [19] to determine the precise threshold of satisfiability is that each constraint is *uniquely extendible*. That is, for each possible assignment to $k-1$ variables of a constraint, there is a unique legal value for the kth variable. The NP-complete problem considered is the generic uniquely extendible constraint satisfaction problem UE-CSP.

In d-UE-CSP, each constraint is over a tuple of variables, each variable must take a value from the domain $\{0, \ldots, d-1\}$, and every constraint is uniquely extendible. Furthermore, we denote the problem (k, d)-UE-CSP if every constraint has size k. Note that k-XOR-SAT is exactly $(k, 2)$-UE-CSP.

It is easy to verify that $(k, 2)$-UE-CSP $\in P$ for all k as every uniquely extendible constraint must be a parity constraint and so the problem reduces to solving a system of linear equations modulo 2. Similarly, $(3, 3)$-UE-CSP $\in P$, since it reduces to solving a set of linear equations modulo 3. However:

Theorem 4 *$(3, 4)$-UE-CSP is NP-complete.*

The proof is a reduction from 3-coloring a graph and is placed in the appendix. We note that $(3, 4)$-UE-CSP is NP-complete even when restricted to inputs where no two constraints intersect on more than one variable.

2.2. The random model

For each appropriate n, m, we define $\Omega_{n,m}^{(k,d)}$ to be the set of (k, d)-UE-CSP instances with m constraints on variables $\{v_1, ..., v_n\}$. We define $U_{n,m}^{(k,d)}$ to be a uniformly random member of $\Omega_{n,m}^{(k,d)}$. When m is defined to be some function $g(n)$, we often write $U_{n,m=g(n)}^{(k,d)}$. As is common in the study of random problems of this sort, we will be most interested in the case where $m = cn$ for some constant c.

We will define c^* precisely in the next section. With that definition, a more formal statement of Theorem 1 is that it is the union of the following 2 lemmas:

Lemma 5 *For every $c < c^*$, $U_{n,m=cn}^{(3,4)}$ is a.s. satisfiable.*

Lemma 6 *For every $c > c^*$, $U_{n,m=cn}^{(3,4)}$ is a.s. unsatisfiable.*

3. The 2-core of the underlying hypergraph

Given an instance F of (k, d)-UE-CSP, we define the *underlying hypergraph* of F to be the k-uniform hypergraph whose vertices are the variables of F and whose hyperedges are those k-sets of variables which occur in the constraints of F.

The *2-core* of a hypergraph is the largest (possibly empty) subgraph which has no vertices of degree less than 2. The 2-core is unique, and it can be found using the following procedure:

> **CORE:** While the hypergraph has any vertices of degree less than 2, choose an arbitrary such vertex and delete it, along with all hyperedges containing it.

It is easy to see that the order in which vertices are chosen to be deleted is irrelevant, in that it does not affect the final output of the procedure. We define the *2-core* of an instance F of (k, d)-UE-CSP to be the instance induced by the 2-core of the underlying hypergraph of F; i.e., the instance formed by the variables and constraints of F whose corresponding vertices and hyperedges are in that 2-core.

Lemma 7 *If F is an instance of (k, d)-UE-CSP, then F is satisfiable iff the 2-core of F is satisfiable.*

Proof Clearly, if the 2-core of F is unsatisfiable then so is F. So assume that the 2-core of F is satisfiable. Consider running CORE on the underlying hypergraph of F, and suppose that the deleted variables are $x_1, x_2, ..., x_t$ in that order. Start with any satisfying assignment of the 2-core. Now restore the deleted variables in reverse order, i.e. $x_t, x_{t-1}, ..., x_1$, each time adding the variable along with the at most one constraint that was deleted when the variable was deleted. Because that at most one constraint is uniquely extendible, there is a value that can be assigned to the variable which does not violate the constraint. This will result in a satisfying assignment for F. $\square$

2-cores of random hypergraphs are well-understood (see, for example, [32, 41]). Given c, define x to be the largest solution to

$$c = \frac{x}{3(1-e^{-x})^2},$$

and define

$$\alpha(c) = \frac{x(1-e^{-x})}{3(1-e^{-x}(1+x))}.$$

From [41] (see also [19] for something a bit weaker but sufficient for our purposes) we can glean the following fact:

Fact 8 *A.s. the 2-core of $U^{(3,4)}_{n,m=cn}$ has $\Theta(n)$ variables and $\alpha(c)+o(1)$ times as many constraints as variables.*

Thus we define c^* to be the solution to $\alpha(c)=1$.

Let $\Psi_{n,m}$ denote the subset of $\Omega^{(3,4)}_{n,m}$ in which every variable lies in at least 2 constraints, and let $U^*_{n,m}$ denote a uniformly random member of $\Psi_{n,m}$.

Fact 9 *For any n, m, n', m', if we condition on the event that the 2-core of $U^{(3,4)}_{n,m=cn}$ has n' variables and m' constraints, then that 2-core is a uniformly random member of $\Psi_{n',m'}$.*

Proof This is an easy variation of the proof of Claim 1 in the proof of Lemma 4(b) from [41], which is itself a very standard argument; we omit the details. □

All this implies that Lemmas 5 and 6 are equivalent to:

Lemma 10 *For every $c<1$, $U^*_{n,m=cn}$ is a.s. satisfiable.*

Lemma 11 *For every $c>1$, $U^*_{n,m=cn}$ is a.s. unsatisfiable.*

The second of these lemmas is straightforward, and we close this section with its proof. The first requires much more work, and we present its proof in the next section.

Proof of Lemma 11 We apply what is, in this field, a very standard and straightforward first moment argument. Consider a random instance F chosen from $\Psi_{n,m=cn}$ and let N denote the number of satisfying assignments of F. We will show that $\mathbf{E}(N)=o(1)$; this implies that a.s. $N=0$; i.e., that a.s. F is unsatisfiable.

Consider any assignment σ of values to the variables of F. The probability that a particular constraint is satisfied by σ is $\frac{1}{4}$. Since there are 4^n choices for σ, this yields

$$\mathbf{E}(N) = 4^n 4^{-m} = 4^{-(c-1)n} = o(1),$$

since $c>1$. □

4. A Second Moment Argument

In this section, we prove Lemma 10, the hardest part of Theorem 1. Inspired by the proof of the corresponding theorem in [19], we apply the second moment argument. Unfortunately, the fact that we have a larger domain size and larger set of constraints to choose from makes these calculations more complicated than those in [19].

As in the proof of Lemma 11, we consider a random instance F chosen from $\Psi_{n,m=cn}$ and let N denote the number of satisfying assignments of F. The main step of this proof will be to compute the second moment of N, obtaining:

Lemma 12 $\mathbf{E}(N^2) = (\mathbf{E}(N))^2(1+o(1))$

Chebychev's Inequality implies:

$$\mathbf{Pr}(N>0) \geq \frac{\mathbf{E}(N)^2}{\mathbf{E}(N^2)},$$

and so Lemma 12 implies Lemma 11.

We remark that unique extendibility is crucial. It is not hard to show that if the underlying constraints of a random problem are not uniquely extendible, then Lemma 12 does not hold for that problem.

We will compute $\mathbf{E}(N^2)$ by putting $\frac{\mathbf{E}(N^2)}{\mathbf{E}(N)^2}$ into the form

$$\int_{a_1}^{b_1}\int_{a_2}^{b_2}\int_{a_3}^{b_3} g(x_1,x_2,x_3)e^{nh(x_1,x_2,x_3)}dx_1\,dx_2\,dx_3$$

where h has a unique maximum in the range of the integrals and then applying the Laplace Method.

Suppose that the 4^n possible assignments are $\sigma_1, ..., \sigma_{4^n}$ and let N_i be the indicator variable that σ_i is a satisfying assignment. Then $N = N_1 + ... + N_{4^n}$ and so, using the fact that $N_i^2 = N_i$, we have $N^2 = N + \sum_{i\neq j} N_i N_j$. Since $N_i N_j = 1$ iff F is satisfied by both σ_i and σ_j, this indicates that we must focus on counting the number of instances satisfied by two different assignments to the variables.

Let σ and τ be arbitrary assignments to the variables, let $\#\mathcal{C}$ be the total number of instances in $\Psi_{n,m}$, and let $\#\mathcal{C}_{\sigma,\tau}$ be the total number of instances in $\Psi_{n,m}$ which are satisfied by both σ and τ. Then,

$$\mathbf{E}(N^2) = \frac{1}{\#\mathcal{C}}\sum_{\sigma\neq\tau}\#\mathcal{C}_{\sigma,\tau}.$$

It is easily seen that there are $(24)^2$ possible uniquely extendible constraints for each triple of variables. We can think of the m constraints as inducing a distribution of $3m$ "places" to the n variables such that each variable receives at least 2 "places". So, $\#\mathcal{C} = (24)^{2m}S(3m,n,2)n!$ where $S(i,j,2)$ counts the number of ways to partition i elements into j sets such that each set has at least 2 elements.

To count the number of instances satisfied by a pair of assignments, consider a triple of variables and 2 assignments to those variables, and count the number of uniquely extendible constraints which are satisfied by both assignments. For all three variables assigned the same value: $\frac{1}{4}(24)^2$, for two of the three variables assigned the same value: 0, for one of the three variables assigned the same value: $\frac{1}{12}(24)^2$, and for all three variables are assigned different values: $\frac{1}{18}(24)^2$.

For each integer a, we define $I_a = \{0, \frac{1}{a}, \frac{2}{a}, \dots, \frac{a-1}{a}, 1\}$. Let $\alpha \in I_n$ be the proportion of variables having the same value in both assignments, let $r \in I_{3m}$ be the proportion of $3m$ "places" in the list of triples that receive one of those αn variables, and let T_k be the number of triples with k of those αn variables.

To enumerate all pairs of assignments, we must count the possible assignments to the variables and count the number of choices for the αn variables. This gives $\sum_{\alpha \in I_n} 4^n 3^{(1-\alpha)n} \binom{n}{\alpha n}$ pairs of assignments.

To enumerate all satisfied instances for one pair of assignments and for each choice of T_0, T_1, T_3, we need to (a) count the ways to choose the triples for T_0, T_1, T_3: $\binom{m}{T_0}\binom{m-T_0}{T_1}$; (b) for each triple, count the number of possible constraints: $\left(\frac{(24)^2}{18}\right)^{T_0} \left(\frac{(24)^2}{12}\right)^{T_1} \left(\frac{(24)^2}{4}\right)^{T_3}$; (c) for each triple in T_1, count the 3 possible positions for the one of those αn variables: 3^{T_1}; (d) finally, distribute the variables amongst the "places".

In total, we have

$$
\begin{aligned}
\mathbf{E}(N^2) &= \frac{1}{(24)^{2m} S(3m,n,2) n!} \\
&\quad \times \sum_{\alpha \in I_n} \sum_{r \in I_{3m}} \sum_{\substack{T_0+T_1+T_3=m \\ T_1+3T_3=3rm}} 4^n 3^{(1-\alpha)n} \binom{n}{\alpha n} \\
&\quad \times \binom{m}{T_0}\binom{m-T_0}{T_1} \left(\frac{(24)^2}{18}\right)^{T_0} \left(\frac{(24)^2}{12}\right)^{T_1} \\
&\quad \times \left(\frac{(24)^2}{4}\right)^{T_3} 3^{T_1} S(r3m, \alpha n, 2)(\alpha n)! \\
&\quad \times S((1-r)3m, (1-\alpha)n, 2)((1-\alpha)n)!.
\end{aligned}
$$

Following the technique of [19], we use the approximation

$$S(i,j,2) \sim \frac{1}{j!}\left(\frac{i}{z_0 e}\right)^i (e^{z_0} - 1 - z_0)^j \Phi(i,j)$$

based on results in [30] where z_0 is the positive real solution of the equation $\frac{j}{i} z_0 = \frac{e^{z_0}-1-z_0}{e^{z_0}-1}$ and where $\Phi(i,j) = \sqrt{\frac{ij}{z_0 j(i-j) - i(i-2j)}}$. The proof is omitted.

Setting $T_3 = tm, T_1 = 3rm - 3tm, T_0 = m - 3rm + 2tm, m = cn$, noting that $T_1 \geq 0$ implies $r \leq t$ and $T_0 \geq 0$ implies $t \geq \frac{3r-1}{2}$, letting $I^r_{cn} = I_{cn} \cap [\frac{3r-1}{2}, r]$, using Stirling's Approximation that $i! \sim i^i e^{-i}\sqrt{2\pi i}$, and simplifying, we obtain:

$$\mathbf{E}(N^2) \sim \sum_{\alpha \in I_n} \sum_{r \in I_{3cn}} \sum_{t \in I^r_{cn}} g(\alpha,r,t) e^{nf(\alpha,r,t)}$$

where

$$
\begin{aligned}
g(\alpha,r,t) &= \Phi(3c,1)^{-1}\Phi(r3c,\alpha) \\
&\quad \times \Phi((1-r)3c,(1-\alpha))(2\pi n)^{-\frac{3}{2}} c^{-1} \\
&\quad \times (\alpha(1-\alpha)(1-3r+2t)(3r-3t)t)^{-\frac{1}{2}} \\
f(\alpha,r,t) &= (2-c-3rc+2tc)\ln 2 \\
&\quad + (1-\alpha-2c+3rc-tc)\ln 3 \\
&\quad - \alpha\ln\alpha - (1-\alpha)\ln(1-\alpha) \\
&\quad - (1-3r+2t)c\ln(1-3r+2t) \\
&\quad - (r-t)3c\ln(r-t) - ct\ln t \\
&\quad + r3c\ln r + (1-r)3c\ln(1-r) \\
&\quad + \alpha\ln(e^z-1-z) - r3c\ln z \\
&\quad + (1-\alpha)\ln(e^y-1-y) \\
&\quad - (1-r)3c\ln y \\
&\quad - \ln(e^x-1-x) + 3c\ln x
\end{aligned}
$$

where $x, y, z > 0$ and

$$
\begin{aligned}
\frac{e^x-1-x}{e^x-1} - \frac{x}{3c} &= \frac{e^y-1-y}{e^y-1} - \frac{y(1-\alpha)}{3c(1-r)} \\
&= \frac{e^z-1-z}{e^z-1} - \frac{z\alpha}{3cr} = 0. \qquad (1)
\end{aligned}
$$

And thus,

$$\frac{\mathbf{E}(N^2)}{\mathbf{E}(N)^2} \sim \sum_{\alpha \in I_n} \sum_{r \in I_{3cn}} \sum_{t \in I^r_{cn}} g(\alpha,r,t) e^{nh(\alpha,r,t)}$$

where $h(\alpha,r,t) = f(\alpha,r,t) - 4(1-c)\ln 2$.

Checking partial derivatives, it is straightforward to verify that $\alpha = \frac{1}{4}, r = \frac{1}{4}, t = \frac{1}{16}, x = y = z$ is a local maximum for f and satisfies (1), and Lemma 16 in the appendix proves that this is, indeed, the only maximum of f on the relevant interval.

Next, we replace the summations with integrals:

$$\frac{\mathbf{E}(N^2)}{\mathbf{E}(N)^2} \sim 3c^2 n^3 \times \int_0^1 \int_0^1 \int_{\max\{0,\frac{3r-1}{2}\}}^r g(\alpha,r,t) e^{nh(\alpha,r,t)} dt\, dr\, d\alpha.$$

The Laplace Method for a triple integral (see, for example, [8] and [17] for descriptions of the method) can be stated as follows.

Lemma 13 *Let*

$$F(n) = \int_{a_1}^{b_1}\int_{a_2}^{b_2}\int_{a_3}^{b_3} g(x_1,x_2,x_3) e^{nh(x_1,x_2,x_3)} dx_1\, dx_2\, dx_3$$

where

(a) *h is continuous in $a_i \le x_i \le b_i$,*

(b) *$h(c_1, c_2, c_3) = 0$ for some point (c_1, c_2, c_3) with $a_i < c_i < b_i$ and $h(x_1, x_2, x_3) < 0$ for all other points in the range,*

(c) *$h(x_1, x_2, x_3) = -\frac{1}{2}\sum_{i=1}^{3}\sum_{j=1}^{3} a_{ij}x_ix_j + o(x_1^2 + x_2^2 + x_3^3)$ with $(x_1^2 + x_2^2 + x_3^2 \to 0)$, and*

(d) *the quadratic form $\sum\sum a_{ij}x_ix_j$ is positive definite.*

Then,

$$F(n) \sim (2\pi)^{\frac{3}{2}} D^{-\frac{1}{2}} n^{-\frac{3}{2}} g(c_1, c_2, c_3)$$

where D is the determinant of the matrix (a_{ij}).

By our choice of h, points (a) and (b) are satisfied. Point (c) is satisfied if we approximate h by the Taylor expansion about the point $\alpha = \frac{1}{4}, r = \frac{1}{4}, t = \frac{1}{16}$ and the a_{ij}'s are from the second partial derivatives of h. This approximation also satisfies (d) and the determinant D of (a_{ij}) is $\frac{2^{15}}{9}Kc$ where $K = \frac{c(e^x-1)^2}{(e^x-1)^2+3c(e^x-xe^x-1)}$. (See the appendix for details.)

Applying the Laplace Method and using the fact that $g(\frac{1}{4}, \frac{1}{4}, \frac{1}{16}) = K^{\frac{1}{2}}2^6 3^{-2} c^{-\frac{3}{2}}(\pi n)^{\frac{3}{2}}$ we obtain

$$\begin{aligned}\frac{\mathbf{E}(N^2)}{\mathbf{E}(N)^2} &\sim 3c^2n^3(2\pi)^{\frac{3}{2}}\left(\frac{2^{15}}{9}Kc\right)^{-\frac{1}{2}} n^{-\frac{3}{2}} g(\frac{1}{4}, \frac{1}{4}, \frac{1}{16}) \\ &\sim 1,\end{aligned}$$

as desired. (See the appendix for a more detailed derivation.) □

5. Resolution complexity

We next give a brief outline of the proof of Theorem 2.

Proof of Theorem 2: The proof follows along the same lines as the techniques of [37] and [42], using the trivial fact that if x is a variable that lies in at most one constraint, then any satisfying assignment for the CSP obtained by deleting x can be extended to a satisfying assignment for the entire formula. We omit the details which should be straightforward to anyone familiar with [37], [42] or one of many similar papers. □

6. Proof of Theorem 3

We close the paper with a proof of Theorem 3. Similar to Section 3, we reduce a formula F with a mixture of clauses of size 2 and 3 to its 2-core. By the same arguments as the proof of Lemma 11, if the 2-core has n' vertices and cn' constraints for $c > 1$, it is a.s. unsatisfiable, and thus F is a.s. unsatisfiable.

Before giving the proof for Theorem 3, we need the following theorem which gives the size of the 2-core of a random hypergraph with a mixture of edges of size 2 and edges of size 3. The proof of the theorem closely follows the proof of the theorem for k-cores in random uniform hypergraphs in [39] and so is omitted.

Theorem 14 *Let $c_2, c_3 \ge 0$. Let x be the largest solution to*

$$x = (1 - e^{-x})^2 3c_3 + (1 - e^{-x})2c_2.$$

If $x > 0$, then a uniformly random hypergraph with c_2n 2-edges, c_3n 3-edges and no other edges has a 2-core with $\alpha(c_2, c_3)n + o(n)$ vertices, $\beta_2(c_2, c_3)n + o(n)$ 2-edges and $\beta_3(c_2, c_3)n + o(n)$ 3-edges where $\alpha(c_2, c_3) = 1 - e^{-x} - xe^{-x}$, $\beta_2(c_2, c_3) = c_2(1 - e^{-x})^2$, and $\beta_3(c_2, c_3) = c_3(1 - e^{-x})^3$.

Proof of Theorem 3: Consider a random d-UE-SAT formula F, $d \ge 2$, on n variables with $\left(\frac{1}{2} - \epsilon\right)n$ constraints of size 2 and $\left(\frac{1}{6} + \delta\right)n$ constraints of size 3 for some $\delta, \epsilon = \epsilon(\delta) > 0$ where δ is arbitrary and ϵ will be chosen later.

Take the underlying hypergraph H of F, and assume H has a 2-core H' with αn vertices and βn hyperedges. Consider the subformula F' of F which corresponds to H'. By the same argument as Fact 9, F' is uniformly random conditional on the number of variables, constraints of size 2 and constraints of size 3, and if we choose an assignment, that assignment satisfies each constraint of F', regardless of the size of that constraint, with probability $\frac{1}{d}$. Thus by the same argument as as the proof of Lemma 11,

$$\begin{aligned}\mathbf{E}(\text{\# of satisfying assignments}) &= d^{\alpha n}\left(\frac{1}{d}\right)^{\beta n} \\ &= o(1) \text{ if } \beta > \alpha.\end{aligned}$$

Thus, if $\beta > \alpha$, F' is a.s. unsatisfiable and so F is a.s. unsatisfiable.

Now we prove F has a 2-core with more edges than vertices by applying Theorem 14 with $c_2 = \frac{1}{2} - \epsilon$ and $c_3 = \frac{1}{6} + \delta$. Lemma 15 proves that for all $\delta > 0$ there exists an $\epsilon > 0$ such that the x of Theorem 14 is positive and $\beta = \beta_2(c_2, c_3) + \beta_3(c_2, c_3) > \alpha(c_2, c_3) = \alpha$. Thus, we pick an ϵ which satisfies Lemma 15 and complete the proof. □

Lemma 15 *For any $\delta > 0$, there exists $\epsilon > 0$ such that the largest solution to*

$$x = (1 - e^{-x})^2 3\left(\frac{1}{6} + \delta\right) + (1 - e^{-x})2\left(\frac{1}{2} - \epsilon\right) \quad (2)$$

is greater than 0 and

$$1 - e^{-x} - xe^{-x} < \left(\frac{1}{2} - \epsilon\right)(1 - e^{-x})^2 + \left(\frac{1}{6} + \delta\right)(1 - e^{-x})^3. \quad (3)$$

Proof. Solving (2) for $(1-2\epsilon)$ gives

$$(1-2\epsilon) = \frac{2x-(1-e^{-x})^2(1+6\delta)}{2(1-e^{-x})} \tag{4}$$

and solving (3) for $(1-2\epsilon)$ gives

$$(1-2\epsilon) > \frac{6(1-e^{-x}-xe^{-x})-(1-e^{-x})^3(1+6\delta)}{3(1-e^{-x})^2}. \tag{5}$$

If the rhs's of (4) and (5) are equal, we have

$$\delta = \frac{x+xe^{-x}+2e^{-x}-2}{(1-e^{-x})^3} - \frac{1}{6}. \tag{6}$$

Let x_δ be the positive solution to (6). Plotting (4) and (5) shows that for all $x > x_\delta$, the $(1-2\epsilon)$ value from the equality (4) always satisfies the inequality (5).

Now, prove that for any $\delta > 0$, there exists $\epsilon > 0$ such that $x > x_\delta$. From (2) we have

$$\delta = \frac{x-(1-e^{-x})(1-\epsilon)}{3(1-e^{-x})^2} - \frac{1}{6}. \tag{7}$$

Consider the δ of (6) as a function of x, $\delta_6(x)$, consider the δ of (7) as a function of x and ϵ, $\delta_7(x,\epsilon)$, and let $\delta_\epsilon(x) = \lim_{\epsilon\to 0}\delta_7(x,\epsilon)$. Plotting $\delta_\epsilon(x)$ and $\delta_6(x)$, we see that for any $x > 0$, $\delta_\epsilon(x) < \delta_6(x)$. If we let $\lim_{\epsilon\to 0} x = \delta_\epsilon^{-1}(\delta)$ and $x_\delta = \delta_6^{-1}(\delta)$, then for any $\delta > 0$, $\lim_{\epsilon\to 0} x > x_\delta$ and thus there exists $\epsilon > 0$ such that $x > x_\delta$. □

References

[1] D. Achlioptas. Lower bounds for random 3-SAT via differential equations. *Theoretical Computer Science*, 265(1–2):159–185, August 2001.

[2] D. Achlioptas, P. Beame, and M. Molloy. A sharp threshold in proof complexity yields lower bounds for satisfiability search. *Journal of Computer and System Sciences*, 68(2):238–268, March 2004.

[3] D. Achlioptas, A. Chtcherba, G. Istrate, and C. Moore. The phase transition in 1-in-k SAT and NAE 3-SAT. In *Proceedings of the Twelfth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 721–722, 2001.

[4] D. Achlioptas, L. M. Kirousis, E. Kranakis, and D. Krizanc. Rigorous results for random $(2+p)$-SAT. *Theoretical Computer Science*, 265(1–2):109–129, August 2001.

[5] D. Achlioptas, M. S. O. Molloy, L. M. Kirousis, Y. C. Stanatiou, E. Kranakis, and D. Krizanc. Random constraint satisfaction: A more accurate picture. *Constraints*, 6(4):329–344, October 2001.

[6] D. Achlioptas and C. Moore. The asymptotic order of the random k-SAT threshold. In *Proceedings of the 43rd Annual IEEE Symposium on Foundations of Computer Science*, pages 779–788, 2002.

[7] T. M. Apostol. *Mathematical Analysis: A Modern Approach to Advanced Calculus*. Addison-Wesley, 1957 and 1974.

[8] C. Bender and S. Orsag. *Advanced Mathematical Methods for Scientists and Engineers*. McGraw-Hill, 1978.

[9] A. Braunstein, M. Mezard, M. Weigt, and R. Zecchina. Constraint satisfaction by survey propagation. arxiv.org/abs/cond-mat/0212451, 2003.

[10] P. Cheeseman, B. Kanefsky, and W. M. Taylor. Where the REALLY hard problems are. In *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence*, pages 331–337, 1991.

[11] V. Chvátal and B. Reed. Mick gets some (the odds are on his side). In *Proceedings of the 33rd IEEE Symposium on Foundations of Computer Science*, pages 620–627, 1992.

[12] V. Chvátal and E. Szemerédi. Many hard examples for resolution. *Journal of the ACM*, 35(4):759–768, October 1988.

[13] S. Cocco, R. Monasson, A. Montanari, and G. Semerjian. Approximate analysis of search algorithms with "physical" methods. arxiv.org/abs/cs.CC/0302003, 2003.

[14] H. Connamacher. A random constraint satisfaction problem that seems hard for DPLL. In *Proceedings of the Seventh International Conference on Theory and Applications of Satisfiability Testing*, pages 3–11, 2004.

[15] N. Creignou and H. Daudé. Crossed classification, complexity versus phase transition, for boolean CSPs. In *Proceedings of the Sixth International Conference on Theory and Applications of Satisfiability Testing*, 2003.

[16] N. Creignou and H. Daudé. Generalized satisfiability problems: Minimal elements and phase transisitons. *Theoretical Computer Science*, 302(1–3):417–430, June 2003.

[17] N. G. de Bruijn. *Asymptotic Methods in Analysis*. North-Holland, 1958 and 1970. Dover reprint, 1981.

[18] O. Dubois, Y. Boufkhad, and J. Mandler. Typical random 3-SAT formulae and the satisfiability threshold. Technical Report TR03-007, Electronic Colloquium on Computational Complexity, 2003.

[19] O. Dubois and J. Mandler. The 3-XORSAT threshold. In *Proceedings of the 43rd Annual IEEE Symposium on Foundations of Computer Science*, pages 769–778, 2002.

[20] M. Dyer, A. Frieze, and M. Molloy. A probabilistic analysis of randomly generated binary constraint satisfaction problems. *Theoretical Computer Science*, 290(3):1815–1828, January 2003.

[21] W. Fernandez de la Vega. On random 2-SAT. Manuscript, 1992.

[22] A. D. Flaxman. A sharp threshold for a random constraint satisfaction problem. *Discrete Mathematics*, 285(1–3):301–305, August 2003.

[23] E. Friedgut. Sharp thresholds of graph properties and the k-SAT problem. *Journal of the American Mathematical Society*, 12(4):1017–1054, 1999.

[24] A. Frieze and N. C. Wormald. Random k-SAT: A tight threshold for moderately growing k. In *Proceedings of the Fifth International Symposium on the Theory and Applications of Satisfiability Testing*, pages 1–6, 2002.

[25] A. M. Frieze and M. Molloy. The satisfiability threshold for randomly generated binary constraint satisfaction problems. In *Proceedings of the 7th International Workshop on Randomization and Approximation Techniques in Computer Science*, 2003.

[26] Y. Gao and J. Culberson. Resolution complexity of random constraint satisfaction problems: Another half of the story. *Electronic Notes in Discrete Mathematics*, 16, 2003.

[27] I. P. Gent, E. MacIntyre, P. Prosser, B. M. Smith, and T. Walsh. Random constraint satisfaction: Flaws and structure. *Constraints*, 6(4):345–372, October 2001.

[28] A. Goerdt. A threshold for unsatisfiability. *Journal of Computer and System Sciences*, 53(3):469–486, December 1996.

[29] M. T. Hajiaghayi and G. B. Sorkin. The satisfiability threshold of random 3-SAT is at least 3.52. arxiv.org/abs/math.CO/0310193, 2003.

[30] F. Hennecart. Stirling distributions and stirling numbers of the second kind. computational problems and statistics. *Kybernetika*, 30(3):279–288, 1994.

[31] A. C. Kaporis, L. M. Kirousis, and E. Lalas. Selecting complementary pairs of literals. *Electronic Notes in Discrete Mathematics*, 16, 2003.

[32] B. S. Majewski, N. C. Wormald, G. Havas, and Z. J. Czech. A family of perfect hashing methods. *The Computer Journal*, 39(6):547–554, 1996.

[33] O. C. Martin, R. Monasson, and R. Zecchina. Statistical mechanics methods and phase transitions in optimization problems. *Theoretical Computer Science*, 265(1–2):3–67, August 2001.

[34] M. Mezard, G. Parisi, and M. A. Virasoro. *Spin Glass Theory and Beyond*, volume 9 of *World Scientific Lecture Notes in Physics*. World Scientific, 1987.

[35] M. Mezard, G. Parisi, and R. Zecchina. Analytic and algorithmic solution of random satisfiability problems. *Science*, 297:812–815, August 2002.

[36] M. Mézard, F. Ricci-Tersenghi, and R. Zecchina. Two solutions to diluted p-spin models and XORSAT problems. *Journal of Statistical Physics*, 111(3–4):505–533, May 2003.

[37] D. G. Mitchell. Resolution complexity of random constraints. In *Principles and Practices of Constraint Programming – CP 2002*, pages 295–309, 2002.

[38] D. G. Mitchell, B. Selman, and H. J. Levesque. Hard and easy distributions of SAT problems. In *Proceedings of the Tenth National Conference on Artificial Intelligence*, pages 459–465, 1992.

[39] M. Molloy. Models and threshold for random constraint satisfaction problems. To appear in SIAM Journal of Computing. Extended abstract in STOC 2002, 209–217.

[40] M. Molloy. When does the giant component bring unsatisfiability? Submitted.

[41] M. Molloy. The pure literal rule threshold and cores in random hypergraphs. In *Proceedings of the Fifteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, 2004.

[42] M. Molloy and M. Salavatipour. The resolution complexity of random constraint satisfaction problems. Preprint. Extended abstract in the Proceedings of FOCS 2003, 330-339, 2003.

[43] R. Monasson and R. Zecchina. Tricritical points in random combinatorics: the $(2 + p)$-SAT case. *Journal of Physics A: Mathematical and General*, 31(42,20):9209–9217, November 1998.

[44] R. Monasson, R. Zecchina, S. Kirkpatrick, B. Selman, and L. Troyansky. Phase transitions and search cost in the $2+p$–SAT problem. In *4th Workshop on Physics and Computation*, 1996.

[45] R. Monasson, R. Zecchina, S. Kirkpatrick, B. Selman, and L. Troyansky. $2 + p$-SAT: Relation of typical-case complexity to the nature of the phase transition. *Random Structures and Algorithms*, 15(3–4):414–435, October–December 1999.

[46] R. Monasson, R. Zecchina, S. Kirkpatrick, B. Selman, and L. Troyansky. Determining computational complexity from characteristic 'phase transitions'. *Nature*, 400:133–137, July 1999.

[47] T. J. Schaefer. The complexity of satisfiability problems. In *Proceedings of the Tenth Annual ACM Symposium on Theory of Computing*, pages 216–226, 1978.

[48] B. M. Smith. Constructing an asymptotic phase transition in random binary constraint satisfaction problems. *Theoretical Computer Science*, 265(1–2):265–283, August 2001.

[49] K. Xu and W. Li. Exact phase transitions in random constraint satisfaction problems. *Journal of Artificial Intelligence Research*, 12:93–103, 2000.

[50] K. Xu and W. Li. Many hard examples in exact phase transitions with application to generating hard satisfiable instances. Technical Report cs.CC/0302001, Computing Research Repository (CoRR), 2003.

A. Proof of Theorem 4

Here we present a proof that $(3, 4)$-UE-CSP is NP-complete.

Proof. Clearly the problem is in NP. We will prove it is NP-hard by showing 3-COLOR $\leq_p (3, 4)$-UE-CSP. For 3-COLOR, we are given a graph G with n vertices and m edges, and we wish to color G with three colors such that any two vertices joined by an edge cannot receive the same color. Now given G, we will create an instance of $(3, 4)$-UE-CSP with $n+m$ variables and $3m$ constraints such that G is 3 colorable if and only if there is a valid assignment to the variables of the CSP. For each edge $e = uv$ of G, we will have three variables, u, v, e, and we will add 3 uniquely extendible constraints to $\{u, v, e\}$. These constraints are listed in Figure 1.

Note that u, v, and e may be set any permutation of $\{0, 1, 2\}$. However, no variable can receive a value of 3 without violating one of the constraints. Likewise, u and v can not be assigned the same value without violating the constraints. If we let the colors for G be $\{0, 1, 2\}$, the proof follows. $\square$

Remark: This proof can be extended to show $(3, 4)$-UE-CSP is NP-complete even when restricted to inputs where no two constraints intersect on more than one variable. We omit the details.

Constraint 1			Constraint 2			Constraint 3		
u	v	e	u	v	e	u	v	e
0	0	0	0	0	3	0	0	3
0	1	2	0	1	2	0	1	2
0	2	1	0	2	1	0	2	1
0	3	3	0	3	0	0	3	0
1	0	2	1	0	2	1	0	2
1	1	3	1	1	1	1	1	3
1	2	0	1	2	0	1	2	0
1	3	1	1	3	3	1	3	1
2	0	1	2	0	1	2	0	1
2	1	0	2	1	0	2	1	0
2	2	3	2	2	3	2	2	2
2	3	2	2	3	2	2	3	3
3	0	3	3	0	0	3	0	0
3	1	1	3	1	3	3	1	1
3	2	2	3	2	2	3	2	3
3	3	0	3	3	1	3	3	2

Figure 1. The constraints used in the proof of Theorem 4. Each row of a constraint lists the ordered triples of values which the constraint permits to be assigned to the 3 variables.

B. Proof of Lemma 12

Here we present more details of the proof of Lemma 12. As justified in Section 4, we have

$$\frac{\mathbf{E}(N^2)}{\mathbf{E}(N)^2} \sim \sum_{\alpha \in I_n} \sum_{r \in I_{3cn}} \sum_{t \in I_{cn}^r} g(\alpha, r, t) e^{nh(\alpha, r, t)}$$

where $h(\alpha, r, t) = f(\alpha, r, t) - 4(1-c)\ln 2$ and

$$\begin{aligned} f(\alpha, r, t) &= (2 - c - 3rc + 2tc)\ln 2 \\ &\quad + (1 - \alpha - 2c + 3rc - tc)\ln 3 \\ &\quad - \alpha \ln \alpha - (1-\alpha)\ln(1-\alpha) \\ &\quad - (1 - 3r + 2t)c\ln(1 - 3r + 2t) \\ &\quad - (r - t)3c\ln(r - t) - ct\ln t \\ &\quad + r3c\ln r + (1 - r)3c\ln(1 - r) \\ &\quad + \alpha\ln(e^z - 1 - z) - r3c\ln z \\ &\quad + (1 - \alpha)\ln(e^y - 1 - y) \\ &\quad - (1 - r)3c\ln y \\ &\quad - \ln(e^x - 1 - x) + 3c\ln x, \qquad (8) \\ g(\alpha, r, t) &= \Phi(3c, 1)^{-1}\Phi(r3c, \alpha) \\ &\quad \times \Phi((1-r)3c, (1-\alpha))(2\pi n)^{-\frac{3}{2}}c^{-1} \\ &\quad \times (\alpha(1-\alpha)(1-3r+2t)(3r-3t)t)^{-\frac{1}{2}}. \end{aligned}$$

To use the Laplace Method to approximate this sum, we must find the maximums for f. If we differentiate f with respect to α, r, t and apply (1), we get

$$\begin{aligned} \frac{\partial f}{\partial \alpha} &= -\ln 3 - \ln\alpha + \ln(1-\alpha) + \ln(e^z - 1 - z) \\ &\quad - \ln(e^y - 1 - y) \\ \frac{\partial f}{\partial r} &= -3c\ln 2 + 3c\ln 3 - 3c\ln(r - t) + 3c\ln r \\ &\quad - 3c\ln(1 - r) + 3c\ln(1 - 3r + 2t) - 3c\ln z \\ &\quad + 3c\ln y \\ \frac{\partial f}{\partial t} &= 2c\ln 2 - c\ln 3 + 3c\ln(r - t) - c\ln t \\ &\quad - 2c\ln(1 - 3r + 2t). \end{aligned}$$

Setting the partial derivatives to 0 implies

$$\frac{1-\alpha}{\alpha} = 3\frac{e^y - 1 - y}{e^z - 1 - z} \qquad (9)$$

$$\frac{1-r}{r} = \frac{3}{2}\cdot\frac{y}{z}\cdot\frac{(1-3r+2t)}{(r-t)} \qquad (10)$$

$$\frac{(r-t)^3}{(1-3r+2t)^2} = \frac{3t}{4}. \qquad (11)$$

From (9), (10) and (11) it is straightforward to verify that $\alpha = \frac{1}{4}, r = \frac{1}{4}, t = \frac{1}{16}$, $x = y = z$ is a local maximum for f and satisfies (1), $f(\frac{1}{4}, \frac{1}{4}, \frac{1}{16}) = 4\ln 2 - 4c\ln 2$, and Lemma 16 in Section B.1 proves that this is, indeed, the only maximum of f on the relevant interval.

Next, we replace the summations with integrals:

$$\frac{\mathbf{E}(N^2)}{\mathbf{E}(N)^2} \sim 3c^2n^3 \times \int_0^1\int_0^1\int_{\max\{0, \frac{3r-1}{2}\}}^r g(\alpha, r, t)e^{nh(\alpha, r, t)}dt\, dr\, d\alpha.$$

To approximate the value of the integrals with the Laplace Method, we will approximate h by the Taylor expansion about the point $\alpha = \frac{1}{4}$, $r = \frac{1}{4}$, $t = \frac{1}{16}$. The second partial derivatives of f are:

$$\begin{aligned} f_{\alpha\alpha} &= -\frac{1}{\alpha} - \frac{1}{1-\alpha} + \frac{e^z - 1}{e^z - 1 - z}\frac{\partial z}{\partial \alpha} \\ &\quad - \frac{e^y - 1}{e^y - 1 - y}\frac{\partial y}{\partial \alpha} \\ f_{\alpha r} &= \frac{e^z - 1}{e^z - 1 - z}\frac{\partial z}{\partial r} - \frac{e^y - 1}{e^y - 1 - y}\frac{\partial y}{\partial r} \\ f_{\alpha t} &= 0 \\ f_{r\alpha} &= \frac{-3c}{z}\frac{\partial z}{\partial \alpha} + \frac{3c}{y}\frac{\partial y}{\partial \alpha} \\ f_{rr} &= -\frac{3c}{r-t} + \frac{3c}{r} + \frac{3c}{1-r} - \frac{9c}{1-3r+2t} \\ &\quad - \frac{3c}{z}\frac{\partial z}{\partial r} + \frac{3c}{y}\frac{\partial y}{\partial r} \\ f_{rt} &= \frac{3c}{r-t} + \frac{6c}{1-3r+2t} \\ f_{t\alpha} &= 0 \\ f_{tr} &= \frac{3c}{r-t} + \frac{6c}{1-3r+2t} \\ f_{tt} &= -\frac{3c}{r-t} - \frac{c}{t} - \frac{4c}{1-3r+2t}. \end{aligned}$$

Where, from (1),

$$\frac{\partial z}{\partial \alpha} = \frac{-z(e^z-1)^2}{\alpha(e^z-1)^2+3rc(e^z-ze^z-1)}$$
$$\frac{\partial z}{\partial r} = \frac{\alpha z(e^z-1)^2}{r[\alpha(e^z-1)^2+3rc(e^z-ze^z-1)]}\frac{\partial y}{\partial \alpha}$$
$$= \frac{y(e^y-1)^2}{(1-\alpha)(e^y-1)^2+3(1-r)c(e^y-ye^y-1)}$$
$$\frac{\partial y}{\partial r} = \frac{-(1-\alpha)y(e^y-1)^2}{(1-r)[(1-\alpha)(e^y-1)^2+3(1-r)c(e^y-ye^y-1)]}.$$

We approximate h by a Taylor expansion about its maximum, and the matrix (a_{ij}) is formed from the value of the second partial derivatives of f at this point. Since $x = y = z$ at the maximum and if we let $K = \frac{c(e^x-1)^2}{(e^x-1)^2+3c(e^x-xe^x-1)}$,

$$(a_{ij}) = \begin{bmatrix} \frac{16}{3}+16K & -16K & 0 \\ -16K & 24c+16K & -32c \\ 0 & -32c & \frac{128}{3}c \end{bmatrix}.$$

Using (1) and the fact $x > 0$ gives $K > 0$ which implies the quadratic form of (a_{ij}) is positive definite (see, e.g., [7] p. 142), and the determinant D of (a_{ij}) is $\frac{2^{15}}{9}Kc$.

Now, we can apply the Laplace Method of Lemma 13 and get,

$$\frac{\mathbf{E}(N^2)}{\mathbf{E}(N)^2} \sim 2^{-6}3^2(\pi cn)^{\frac{3}{2}}K^{-\frac{1}{2}}g(\frac{1}{4},\frac{1}{4},\frac{1}{16}).$$

Note that $\Phi(vi, vj) = \Phi(i,j)$ for any value v. Using this fact, we can simplify $g(\frac{1}{4},\frac{1}{4},\frac{1}{16})$.

$$g(\frac{1}{4},\frac{1}{4},\frac{1}{16}) = \Phi(3c,1)2^6 3^{-2}c^{-1}(\pi n)^{-\frac{3}{2}}. \tag{12}$$

From (1), $e^x = 1 + \frac{3cx}{3c-x}$, we can simplify K.

$$K = c(\Phi(3c,1))^2. \tag{13}$$

Thus,

$$\begin{aligned} \frac{\mathbf{E}(N^2)}{\mathbf{E}(N)^2} &\sim \Phi(3c,1)c^{\frac{1}{2}}K^{-\frac{1}{2}} && \text{by (12)} \\ &\sim 1. && \text{by (13)} \end{aligned}$$

B.1. Proof that f has only one maximum

Lemma 16 *The only maximum of (8) when $\alpha \in [0,1]$, $r \in [0,1]$, $t \in [\max\{0, \frac{3r-1}{2}\}, \min\{1,r\}]$ occurs where $y = z$.*

Proof. Let $A = e^z - 1$ and let $B = e^y - 1$. Now, we combine the partial derivatives of f with (1) to create a single polynomial.

From (10),

$$t = \frac{2zr(1-r)-3yr+9yr^2}{6yr+2z(1-r)}. \tag{14}$$

From (1),

$$r = \frac{z\alpha A}{3c(A-z)}. \tag{15}$$

From (9),

$$\alpha = \frac{A-z}{A-z+3B-3y}. \tag{16}$$

From (1) plugging in (15) for r and (16) for α,

$$A = \frac{3cz+3By-9cB+9cy}{3c-z}. \tag{17}$$

From (11) and plugging in (14) for t, (15) for r, (16) for α, and (17) for A gives a polynomial of degree 3 in z which we place into canonical form.

$$\begin{aligned} &z^3[6cB-2B^2] \\ &\quad + z^2[9c^2y+18cyB+9yB^2-18c^2B-6cB^2] \\ &+ z[54c^2y^2+18cBy^2-108c^2yB-36cyB^2+36c^2B^2] \\ &\quad + 81c^2y^3+54cy^3B+9y^3B^2-108c^2y^2B \\ &\qquad - 54cy^2B^2+162c^2yB^2 = 0. \end{aligned}$$

Since we know there is a solution when $z = y$, we will divide out the root $(z-y)$ which gives the polynomial

$$\begin{aligned} &z^2[6cB-2B^2] \\ &\quad + z[9c^2y+24cyB+7yB^2-18c^2B-6cB^2] \\ &\quad + 63c^2y^2+42cy^2B+7y^2B^2 \\ &\qquad - 126c^2yB-42cyB^2+36c^2B^2 \end{aligned}$$

and the remainder

$$\begin{aligned} &144c^2y^3+96cy^3B+16y^3B^2-288c^2y^2B \\ &\quad -96cy^2B^2+144c^2yB^2. \end{aligned}$$

Since $(z-y)$ is a root, we can set the remainder to 0 and solve for y which gives

$$y = \frac{3cB}{3c+B}.$$

Now, we can plug the value for y back into (17) which gives

$$z = \frac{3cA}{3c+A}.$$

Thus, we must have y=z. $\square$