Well out of reach: Why hard problems are hard*

Joseph Culberson[1] and Ian P. Gent[2]

[1] Department of Computing Science, University of Alberta, Edmonton, Alberta, Canada, T6G 2H1. joe@cs.ualberta.ca
[2] Department of Computer Science, University of Strathclyde, Glasgow G1 1XH, United Kingdom. ipg@cs.strath.ac.uk

Report APES-13-1999

June 1999

Abstract. We show that problems at the uncolorability phase transition are well out of reach of intelligent algorithms. Since there are not small and easily checkable subgraphs which can be used to confirm uncolorability quickly, we cannot hope to build more intelligent algorithms to avoid hard problems at the phase transition. Also, our results suggest that a conjectured double phase transition in graph coloring occurs only in small graphs. Similar results are likely in other NP-complete problems where instances from phase transitions are hard for all known algorithms, and will help to explain the phenomenon. Furthermore, our results help to elucidate the distinction between polynomial and non-polynomial search behavior.

This paper is an extended version of a paper submitted to *CP-99*

Available from http://www.cs.strath.ac.uk/~apes/apesreports.html

* Joseph Culberson's work was supported in part by an NSERC grant OGP8053 and an EPSRC visiting fellowship, GR/M54605. We thank members of APES at Leeds and Strathclyde for their help with this work, especially Isla Ross and Toby Walsh.

1 Introduction

There has been a wealth of research showing that a phase transition in solvability of NP-complete problems is often correlated with a peak in hardness of solving those problems. We attempt to understand why phase transitions so often yield hard instances. We do so in the context of graph coloring.

Our main conclusion is that hard instances are well out of reach of even intelligent algorithms. Intelligent algorithms can take account of small local structures in instances to reduce search. For example, our graph coloring algorithm identifies 4-cliques as making a graph non 3-colorable. However, we have found that at phase transitions, the structures which cause unsolvability are very large. In graph coloring, when there are not 4-cliques in graphs, there are often hundreds of critical edges which must be used in proving unsolvability. It is therefore unimaginable to adapt an algorithm to check for them explicitly, suggesting that hardness at phase transitions is indeed an algorithm independent property. We expect that similar behavior will be seen in many other NP problems. For example, in SAT it is known that resolution proofs become large in the overconstrained region [4]. It is likely that a similar result also holds at the phase transition. We conjecture that NP problems with hard phase transitions are those in which the smallest structures responsible for unsolvability are well beyond the reach of reasonable algorithms.

We also investigate another important point in the phase transition, the 'first frozen' point. In coloring, this is the first point at which two nodes are frozen together, i.e. must be colored the same way in all valid colorings. It also is the first time that a search algorithm can make a mistake at its first branching point by setting two nodes to different colors. The possibility of early mistakes has been associated with exceptionally hard instances (ehi's). It has been further suggested that ehi's are responsible for a second phase transition, from polynomial to exponential average case [8]. However, our results suggest that the first frozen point converges on the threshold in colorability and is also similar to the threshold in other ways. Again we expect that similar behavior will be seen in many other NP problems, including those like SAT and constraint satisfaction in which ehi's are well studied.

To support our conclusions we look at critical subgraphs of unsolvable graphs: that is a minimal uncolorable subgraph. We observe that an instance cannot be hard if it is unsolvable for reasons detected by an algorithm in polynomial time: in the case of coloring this occurs when instances contain a small critical subgraph. As we increase the number of nodes, we see that small critical graphs disappear, and we see a huge jump in the size of critical graphs. The result is that more intelligent algorithms cannot be expected to perform significantly better, unless some remarkable new method for proving uncolorability with structures other than subgraphs is discovered. Thus phase transition instances can be expected to be hard for all algorithms.

Monasson et al have studied the change from polynomial to non-polynomial search cost in SAT [12]. We suggest that future studies of this kind can use the existence or not of small reasons for unsolvability as an indicator of whether a sudden jump from polynomial to non-polynomial behavior can be expected.

After background material we describe the techniques we use to identify the threshold in colorability and frozen pairs of nodes. We then describe three empirical studies which support our conclusions. These are for 4-coloring random graphs, 3-coloring random graphs, and 3-coloring triangle-free graphs. We discuss related and future work before concluding.

2 Background

Many problems exhibit the characteristics of a phase transition: as the number of variables goes to infinity, there is a threshold density above which the probability of an instance being insoluble goes to zero, below which the probability goes to one. In graph coloring and other NP problems, Cheeseman et al pointed out the correlation between a phase transition in the solvability of random problems, and the occurrence of difficult instances for search algorithms [3]. In many problems, no algorithm has been found which eliminates hard instances at phase transitions. Such instances are often used for benchmarking in domains such as satisfiability [11] and constraint satisfaction problems [14, 16]. However there is no simple correlation between hard instances and phase transitions. For example, there is a classic phase transition in the solvability of random Hamiltonian cycle instances [9] but hard instances do not seem to be found there [19].

A secondary effect has been noted in many domains. Underconstrained and usually solvable instances are typically easy but occasional ones are exceptionally hard instances. This has been seen in graph coloring [8], satisfiability [5], constraint satisfaction problems [17] and quasigroup completion [7]. Hogg and Williams suggested that the occurrence of ehi's might be associated with a second phase transition, in this case a transition from polynomial to exponential average search cost [8].

In general, graph coloring is the problem of partitioning the set of vertices of a graph $G = (\mathbf{V}, \mathbf{E})$ into subsets called *color classes*. Typically, the classes are given labels, called *colors*, from the set $\{1 \ldots k\}$. A vertex v in class i under some coloring c is said to have color $c[v] = i$. A coloring c is *legal* if for each edge $(u, v) \in \mathbf{E}$, the vertices of the edge fall in distinct color classes; i.e. $c[u] \neq c[v]$. For the remainder of the paper, unless it is otherwise made clear, we refer only to legal colorings. We use $\mathbf{C} = \mathbf{C}_k(G)$ to refer to the set of all legal k-colorings of a graph G. The decision version, which is NP-complete, gives us k and requires us to determine whether or not G is k-colorable, that is does $\mathbf{C}_k(G) \neq \emptyset$.

Given an uncolorable graph, we will be interested in the reasons for uncolorability, and in particular critical subgraphs and critical sets of edges. A critical subgraph of G is a subgraph which is uncolorable, but which becomes colorable if any edge is removed from it. The critical set of edges in G is the set of edges that occur in every critical subgraph of the graph. Thus, these act as a lower bound on the size of the smallest critical subgraph, although the smallest critical subgraph can be arbitrarily larger than the critical set of the graph. Note that an edge is in the critical set iff its removal from G makes G uncolorable. This gives a straightforward way of computing the critical set.

3 Threshold in Colorability and Frozen Pairs

By taking many samples it can be determined empirically at what density of constraints the problems become 50% unsolvable, and this point is usually considered to represent the threshold, even when no phase transition has been proven to exist.

For problems such as SAT the threshold in satisfiability is associated with a sharp rise in the number of variables that have a single value under all satisfying assignments for the soluble instances [13, 12]. Such variables are called *frozen* variables, and the single value available to a variable is its frozen value. The instance becomes unsatisfiable exactly when a clause is added which requires that at least one of these variables takes a value different than its frozen value.

In graph coloring, due to symmetry under relabeling of the color classes, vertices (the variables of the problem) are never frozen to any value or color in any colorable instance. Instead a solvable instance becomes unsolvable exactly when an edge is added to a pair of vertices that are always the same color in every k-coloring of the instance. We refer to such a pair in a solvable instance as a *frozen same pair*. We can restate this by saying that a pair is frozen the same if under all colorings the pair are always in the same color class of the partition; formally u and v are frozen the same when $c[u] = c[v], \forall c \in \mathbf{C}$. We observe a sharp rise in the number of frozen same pairs as we near this threshold: however in this study we restrict ourselves to studying the first frozen pair.

Note that it is also possible that a pair of vertices are in distinct partition elements under every coloring of a soluble instance. This is trivially the case for edges; the more interesting case is for non-edge pairs. We refer to these pairs as *frozen different*. Adding an edge for a frozen different pair has no impact on the set of colorings of the graph.

For purposes of this study, we consider a model based on the set of $\binom{n}{2}$ pairs of vertices $\{(u, v), u, v \in \mathbf{V}\}$ where the vertices of a pair are unordered and distinct and $n = |V|$. We generate at random one of the $\binom{n}{2}!$ permutations of the set of pairs, calling this the *input sequence*. We build a graph of m edges by choosing the first m pairs from the input sequence. For a given sequence we can determine (using binary search) in $O(\log n)$ coloring attempts the smallest value m^* such that the graph on the first m^* edges is not k-colorable. We define the *threshold* $T(n)$ as the average over the set of all input sequences Π of the minimum values m^*.

$$T(n) = \frac{1}{\binom{n}{2}!} \sum_{\pi \in \Pi} m^*(\pi)$$

Note that $T(n)$ is well defined for all n, although not necessarily easy to compute. Using this model gives several advantages in studying the phase transition, including an expected reduction in variance on computing $T(n)$ for a given sample size. One of the conjectures on the coloring phase transition says that for each k, $T(n)/n$ converges to a constant α_k as $n \to \infty$.

We can also compute the smallest value m^f of m such that there is some pair (x, y) of vertices (necessarily with index in the sequence greater than m^f) frozen the same. There may be more than one pair which are forced to be the same color by the addition of the m^fth edge. We call m^f the first edge causing a frozen same pair, or the *first frozen* for short. We define

$$F(n) = \frac{1}{\binom{n}{2}!} \sum_{\pi \in \Pi} m^f(\pi)$$

The computation of m^f is also based on binary search. Starting from the candidate value of m^f, we add each edge in turn. If this causes uncolorability the pair of nodes were frozen, and we can next try a smaller value of m^f. If the graph is still colorable, we remove the edge and add the next edge: if no edge can be added to force uncolorability we have to try a larger value of m^f. This involves many coloring attempts, although some can be saved. First, when we find a coloring in each attempt, we can remove from consideration any future edge between nodes given different colors in that coloring. Second, when we reduce the candidate value of m^f, we need only continue testing from the frozen pair we found, since removing edges cannot make any more pairs frozen. (In some cases we read off the first frozen points from fuller data exploring the full frozen development.)

Our reason for studying the first frozen point m^f is its correlation to the possible existence of the 'double phase transition'[8] and the appearance of rare *exceptionally hard instances* [17, 5]. This is because it is only when we are at or beyond m^f that a search algorithm can make a single early mistake by setting two nodes different which should be the same, and then start thrashing. Our results suggest that the double phase transition is an effect that will only be seen at small problem sizes. However it remains possible that algorithms could thrash due to making a number of decisions which together make a graph uncolorable.

Clearly, for every sequence π, $m^f(\pi) \leq m^*(\pi)$. Thus, $\gamma = \lim_{n\to\infty} F(n)/n$ (assuming it exists) must be bounded above by α. The evidence we present here strongly suggests that as $n \to \infty$, $\gamma \to \alpha$, and possibly even that the absolute difference between $T(n)$ and $F(n)$ converges. While it might exist for small n, the double phase transition appears to be converging to one phase transition.

Throughout this paper we report results using a backtracking graph coloring program developed by the first author.[3] This program is tuned to perform well on coloring problems with small numbers of colors: in this paper we report results on 3- and 4-coloring. On such problems we find the program to be competitive with the state of the art algorithms.

[3] Unfortunately at the time of writing no report on this algorithm is available.

4 Four coloring Random graphs

We start our empirical investigations with a classic problem, that of 4-coloring random G_{nm} graphs. These are random graphs with n vertices and exactly m edges chosen without restriction.

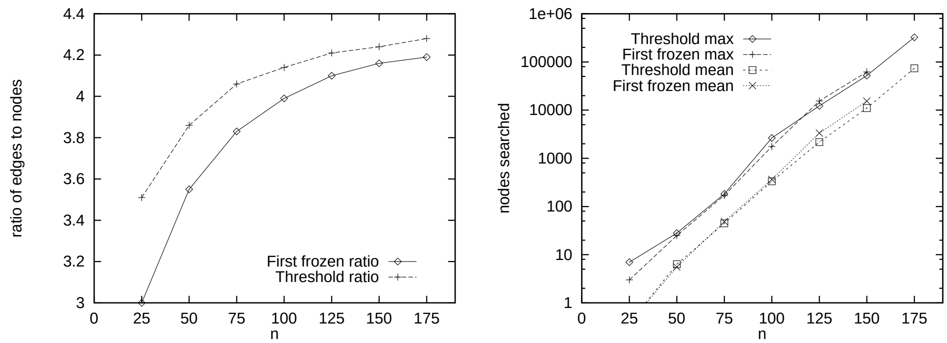

Fig. 1. Comparison of 4-colorability of random graphs at first frozen point and the threshold in colorability. Left: ratio of mean number of edges to n. Right: search cost for uncolorable graphs immediately after addition of critical edge. 50 samples at each n (except 175 for first frozen, where the sample is only 27.)

Our experiments strongly suggest that as n increases, there is very little difference between first frozen and threshold graphs. Fig. 1 shows the ratios at which the threshold and first frozen points occur. Interestingly, the first frozen ratio $F(n)/n$ is converging to the critical threshold $T(n)/n$. closing the gap from a difference of 0.51 at $n = 25$ to 0.09 at $n = 175$. We note that the samples in the two cases are not from the same set of graphs, which might slightly increase the variance in the difference between the mean ratios. Indeed, it is possible that the absolute difference $T(n) - F(n)$ is convergent: we saw the difference vary between 12 and 18 edges, with no obvious increasing or decreasing pattern. We also show the difficulty of coloring these graphs in Fig. 1. We report statistics on the number of search nodes, the most reliable measure of difficulty for the program we are using. At the first frozen point, we use the uncolorable graph in which an edge is added between the two nodes frozen the same. At the threshold we include the graph with the critical edge that makes it uncolorable. The cost in all cases is clearly increasing at an exponential rate. The threshold and first frozen non-colorable instances seem to be almost indistinguishable in difficulty. There seems to be no evidence that the first frozen graphs are exceptionally hard, as there is no systematic difference between the maximum cost at the two points. As expected (but not shown here), the colorable instances without the critical edges are easier, with the graphs at the first frozen being much easier since there are more colorings there.

To support our main conjecture, we need to know if there are small reasons for uncolorability of either first frozen or threshold graphs. To do so we study the critical subgraphs and critical sets of edges in random graphs. The importance of critical subgraphs to the search process is that any search must examine every edge in at least one critical subgraph. While it is not necessarily true that this need be an expensive examination, in NP we would expect that search must be exponential in the size of the smallest critical subgraph. Unfortunately we are not currently able to find the smallest critical subgraphs, though we hope to do so in the future in collaboration with Isla Ross. However, we are able to find a single critical subgraph of each graph, and the critical set of edges in a graph.

	Critical Subgraphs at First Frozen					Critical Subgraphs at Threshold					Difference
n	Mean	Std	Mean/n	Min	Max	Mean	Std	Mean/n	Min	Max	Mean/n
25	26.36	18.81	1.05	10	73	40.02	17.09	1.60	10	74	0.55
50	88.12	55.28	1.76	10	168	119.12	38.02	2.38	10	168	0.62
75	198.44	80.72	2.65	10	268	233.10	39.44	3.11	10	270	0.46
100	310.44	103.77	3.10	10	382	346.52	24.59	3.47	294	395	0.37

Table 1. The number of edges in random critical subgraphs at the first frozen(top) and the number of edges in random critical subgraphs at the threshold for $k = 4$ sample of 50 at each n.

We first study critical subgraphs of the uncolorable graphs at first frozen and threshold points. We can do this only up to $n = 100$ because of the computational expense. The size of these graphs grows at least linearly with n at both points, suggesting that critical graphs have size $O(n)$. As n increases in Table 1, we see that the sizes of critical subgraphs at the first frozen point and threshold are converging. Indeed, comparison of the distribution of the sizes of graphs shows that the convergence is stronger than the table suggests. In figure 2 we compare the histograms of the sizes of critical subgraphs found in the non-4-colorable graphs at the first frozen index with the first encountered frozen pair added, and the threshold graph. These come from 50 samples at $n = 100$. Note that the distributions are very similar, except for 5 graphs with 10 edges, i.e. 5-cliques, in the first frozen instances. This seems to indicate that the critical subgraphs in either case are mostly the same, with the exception that occasionally there may occur a near clique while n remains small. A near-5-clique (n5c), a 5-clique with one missing edge, is the smallest subgraph[4] that can force a pair of vertices to

[4] Smallest as measured by either v or e. We evaluate the expected occurrences of n5c's as the last edge of the critical subgraph is forced.

be the same color under all 4-colorings. We note that if we discount the 5 critical subgraphs of size 10 at $n = 100$ in the first frozen instances, then the average ratio is 3.43, very close to the 3.47 value for the threshold critical graphs.

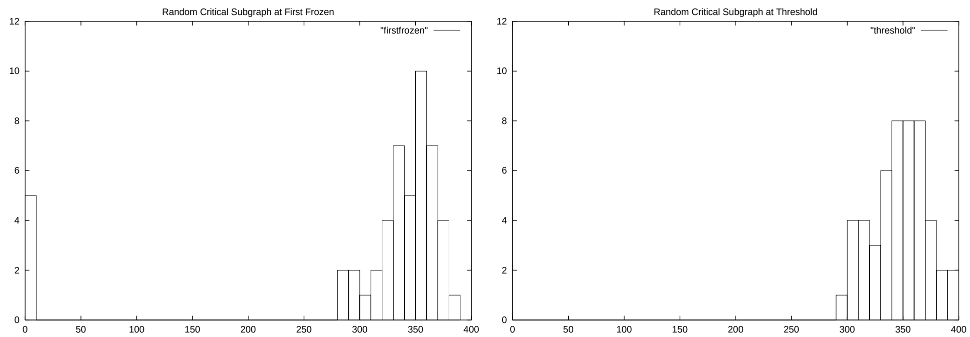

Fig. 2. Comparison of distribution of critical graph sizes on $n = 100, k = 4$ at the first frozen pair (left) and the critical index (right), 50 samples each.

	Critical Sets at First Frozen					Critical Sets at Threshold					Difference
n	Mean	Std	Mean/n	Min	Max	Mean	Std	Mean/n	Min	Max	Mean/n
25	19.90	11.48	0.80	7	52	17.26	10.98	0.69	2	44	-0.11
50	51.38	33.57	1.03	10	117	29.88	21.36	0.60	2	90	-0.43
75	102.10	49.36	1.36	10	199	50.16	36.01	0.67	2	137	-0.69
100	163.12	72.47	1.63	10	316	69.80	53.64	0.70	4	239	-0.93

Table 2. The number of critical edges in first frozen graphs(top) and the number of critical edges in the threshold graphs for $k = 4$ sample of 50 at each n.

We have been reporting the size of an arbitrary critical subgraph, the first that we find. This makes possible in principle that some very small critical subgraphs exist which our procedure does not find. However, this is not the case in 4-coloring. The size of critical sets of edges grows linearly in both cases, as seen in Table 2, and every critical edge must be in every critical subgraph. The smaller size of critical sets at the threshold compared to the first frozen point is because the extra edges create new critical subgraphs, thereby reducing the the intersection of all critical subgraphs. Table 2 also adds further support to our conclusion that large graphs must be searched to prove uncolorability. Since *every* search must look at every edge in the critical set, we cannot hope that more intelligent algorithms will be able to prove graphs uncolorable easily. [5]

The tables and histograms we have presented show that we have found only two kinds of critical subgraphs: small cliques or near-cliques, and large graphs growing in size linearly with n. In fact, straightforward analysis shows that the small graphs disappear as n increases. We can estimate the expected number of near-5-cliques where the edge probability is p by $E[\#n5c] = \binom{n}{5}p^9(1-p)10$ Using $p = e/\binom{n}{2}$, with the empirical $e = 398.56$ as the average number of edges at the first frozen index with $n = 100$ this expectation is 0.0985, which when multiplied by the 50 samples gives us 4.92. [6] Under the assumption that the critical threshold ratio converges to a constant, the expected number of n5c's at the critical threshold (and at the first frozen) is bounded by $O(1/n^4)$ and so must tend rapidly to zero as $n \rightarrow \infty$.

It is not possible that near-5-cliques will be replaced in making graphs uncolorable by other small subgraphs. Given a graph G, the exact formula for $E[G]$ in random graphs depends on the automorphisms of G, but it will be $O(1/n^{e-v})$ where G contains v vertices and e edges. Every node in a critical subgraph must have at least degree k, the number of colors. In this case $k = 4$, so even at the first frozen point where we add the final edge to produce uncolorability, $e \geq 2v - 1$. So any specific larger graph than n5c disappears even faster. Even families of bounded size critical subgraphs disappear, since there is only a finite number of graphs of a given size, each one eventually becoming vanishingly improbable. Our empirical evidence shows that even at $n = 100$, n5c is the only small graph which occurs even occasionally. Taken together, this analysis and our experiments show that uncolorability at either the threshold or first frozen point occurs because of critical subgraphs of size $O(n)$.

To summarise, we have given strong evidence to support two conclusions. First, that problems at the uncolorability phase transition are well out of reach of intelligent algorithms. Since there are not small and easily checkable subgraphs which can be used to confirm uncolorability quickly, we cannot hope to build more intelligent algorithms to avoid hard problems at the phase transition. Second, that there is little difference between graphs at the point where the first pair of nodes become frozen, and at the threshold in colorability. This suggests that there is not a double phase transition in four-coloring random graphs.

---

[5] **Cautionary note:** The first frozen critical graphs are canonical, while the threshold critical graphs are random subgraphs. Thus, there may be correlations in the first frozen critical graphs not present in the threshold critical subgraphs. This does not affect the other measures.

[6] Note that this is not necessarily as indicative as it appears. It is quite possible that the larger critical graphs have an n5c embedded in them, but that the first frozen pair chosen to make the graph non-4-colorable was not the missing edge of the n5c. Also note that e is not a fixed value but is the first frozen point: so that the expectation at a fixed value of e is not the same as the expectation at the first frozen point.

5 Three coloring Random graphs

We also experimented on 3-coloring instances. Figure 3 shows the ratios of the threshold and first frozen points. For the non-3-colorable subgraphs the threshold ratio $T(n)/n$ seems to be converging towards the standard value of 2.3, equivalent to an average degree of 4.6. The size of the first frozen uncolorable graphs appears to be increasing, but it is not clear if convergence is towards the same bound. We also plot the search cost at these points, and again there is no evidence that problems at the first frozen point produce exceptionally hard instances.

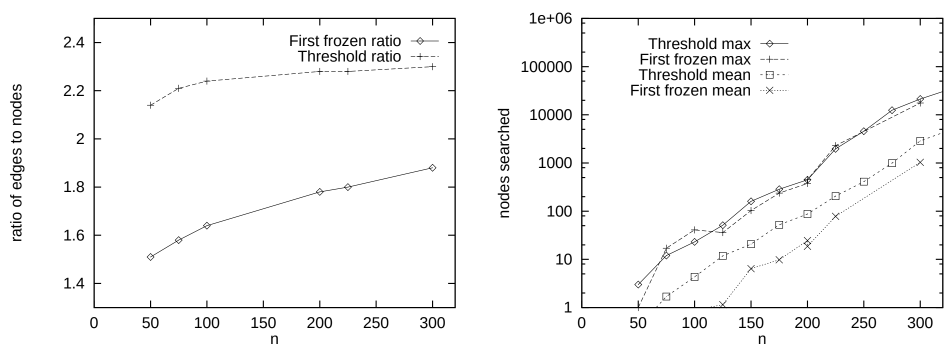

Fig. 3. Comparison of 3-colorability of random graphs at first frozen point and the threshold in colorability. Left: ratio of mean number of edges to n. Right: search cost for uncolorable graphs immediately after addition of critical edge.

There is a much larger gap between first frozen and threshold points in 3-coloring than in 4-coloring. This is partly explained by the smaller critical subgraphs that occur in 3-coloring. The uncolorable 4-clique has 6 edges and 4 nodes. The near-4-clique (n4c), that is the 4-clique with one edge missing, has 5 edges. Thus, the expected number of these in a graph with edge probability p is $E[\#n4c] = \binom{n}{4}p^5(1-p)6$ where $p \approx \frac{2.3n}{\binom{n}{2}}$. Thus, the expected number grows as $O(1/n)$. Eventually the expectation converges to 0. The convergence is much slower than the $O(1/n^4)$ in 4-coloring, and this is responsible for the less clearcut behaviour of 3-coloring. For example, in 300 node graphs, 78 of first frozen pairs were caused by an n4c out of a sample of 100. Nevertheless, our conclusions from the previous section are still supported. Of the remaining 300 node graphs, one contained a critical subgraph with fewer than 12 edges, but the *minimum* size of critical subgraphs in the remaining 21 instances was 407 edges. The empirical evidence suggests that as the 4-clique disappears as the critical subgraph of the first frozen, then the size of the critical subgraphs jumps to $O(n)$ edges. Data suggesting this is shown in Table 7 in the Appendix.

The conclusions we can draw from these experiments and analysis in 3-coloring are the same as in 4-coloring. Asymptotically, we do not expect to see a double phase transition or small uncolorable subgraphs at the phase transition. However, the decay of the near-4-cliques is only $O(1/n)$ and they are still important at $n = 300$. This does mean that it is impossible for us to experiment on large enough graphs from the G_{nm} distribution in which near-4-cliques disappear to confirm our claims.

6 Triangle-free Graphs

Finally, we studied small 3-coloring problems with structure more like those of large graphs after near-4-cliques have disappeared. Analysis shows that asymptotically, graphs at the phase transition have only $O(1)$ triangles, and for simplicity we ban triangles completely and generate triangle-free graphs. The graph coloring problem generators at http://www.cs.ualberta.ca/~joe allow for this, by generating girth 4 graphs. The girth $g(G)$ of a graph G is the length of the smallest cycle, and is 3 if a graph contains a triangle. The reason for having this feature is that algorithms such as DSATUR depend on finding large cliques, or clique-like regions that restrict the colorings allowed locally in the graph. With girth $g > 3$ the largest clique is a 2-clique, or edge. Many heuristics and algorithm designs are thus thwarted. Girth 4 graphs seem to be harder for most programs than unrestricted random graphs.

Girth inhibited graphs are generated as follows. The set of vertex pairs is randomly ordered, and then edges are added in order of this permutation. When an edge x, y is added to the graph, all paths including the edge x, y of length up to $g - 1$ are checked. Any non-edge vertex pair (u, v) in any path is deleted from the set of vertex pairs remaining to be selected. For this experiment, no hidden coloring was given. (The generator also allows the variance in vertex degrees to be restricted, but this was avoided by choosing the maximum delta.) The edge density (probability) was set to 1.0, which means that in each case a maximally dense girth $g = 4$ graph was produced. The algorithm is complete, in that any $g \geq 4$ graph can be generated, but it is unknown if it is uniform: that is we do not claim that each triangle-free graph is equally likely. Once generated, a procedure lists the edges of the graph in random order, and this is the permutation sequence input to our various tests. For small n there is a chance that the graph is three colorable, but this never occurred for the range tested here.

Our experiments on 3-coloring triangle-free graphs give very similar results to the 4-coloring results presented earlier. As n increases, there is very little difference between first frozen and threshold graphs. [7] Fig. 4 shows the ratios at which the threshold and first frozen points occur. Once again, we see threshold ratio $T(n)/n$ and first frozen ratio $F(n)/n$ converging. Again it is possible that even $T(n) - F(n)$ converges, since at $n = 100$ the difference is 23, but by $n = 300$ it has fallen to 18. We also show the difficulty of coloring these graphs in

[7] An interesting aside is that the threshold ratio starts high and goes down.

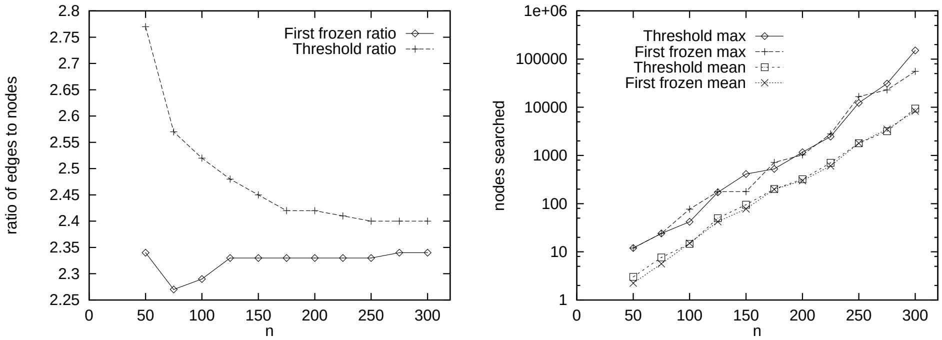

Fig. 4. Comparison of triangle-free graphs at first frozen point and the threshold in colorability. Left: ratio of mean number of edges to n. Right: search cost for uncolorable graphs immediately after addition of critical edge.

Fig. 4. The cost in all cases seems to be increasing at an exponential rate, usually doubling or more for each increase of 25 vertices for the non-colorable instances. The threshold and first frozen non-colorable instances seem to be almost indistinguishable in difficulty. There seems to be no evidence that the first frozen graphs are exceptionally hard, as there is no systematic difference between the maximum cost at the two points.

	Critical Subgraphs at First Frozen					Critical Subgraphs at Threshold					Difference
n	Mean	Std	Mean/n	Min	Max	Mean	Std	Mean/n	Min	Max	Mean/n
50	62.18	16.77	1.24	21	93	76.26	11.74	1.53	46	98	0.29
75	99.50	20.66	1.33	54	138	114.58	18.79	1.53	74	149	0.20
100	148.52	25.28	1.49	86	192	160.92	22.80	1.61	102	201	0.12
125	206.46	29.26	1.65	113	260	217.82	20.49	1.74	154	253	0.09

Table 3. Number of edges in Canonical Critical Subgraphs of the triangle-free graphs. The canonical critical graph is obtained by deleting as many edges as possible choosing them in vertex order.

The size of critical sets and subgraphs are shown in tables 3 and 4. We can do this only up to $n = 125$ because of the computational expense. As n increases in Table 3, we see that the size of critical subgraphs converges at the first frozen point and threshold. The size of these graphs grows at least linearly with n, suggesting that critical graphs have size $O(n)$. The size of critical sets in Table 4

	Critical Sets at First Frozen					Critical Sets at Threshold					Difference
n	Mean	Std	Mean/n	Min	Max	Mean	Std	Mean/n	Min	Max	Mean/n
50	33.90	14.69	0.68	11	83	17.50	13.52	0.35	1	60	-0.33
75	60.32	22.76	0.80	21	108	27.04	20.17	0.36	1	91	-0.44
100	83.70	29.19	0.84	30	139	32.76	24.34	0.33	4	89	-0.51
125	112.42	45.21	0.90	17	203	44.42	28.47	0.36	3	128	-0.54

Table 4. Number of edges in the critical sets of triangle free graphs

is also growing linearly at both points.

To summarise, empirical evidence was not clear in the case of 3-coloring random graphs, so we studied triangle-free graphs, which are similar in structure to large random graphs. The evidence supports our conclusions that hard instances at phase transitions are well out of reach of intelligent algorithms, and that the double phase transition merges into one.

7 Related and Further Work

Exceptionally hard instances (ehi's) are usually understood to arise as the result of a mistake early in the search process: for example a single incorrect variable setting can lead to an unsolvable subproblem using much search, any other setting leading to an easily solved problem [5]. Baker identified this as a thrashing process [1] and Smith and Grant were able to predict with some accuracy the frequency and search cost of ehi's for a very simple backtracking algorithm [18]. Unlike hard instances at phase transitions, ehi's are highly algorithm dependent, and so the distribution of search cost over runs with different heuristics is important [8, 7].

To understand both hard instances at phase transitions and ehi's, a number of authors have studied 'minimal unsatisfiable subproblems' (MUS's) [6, 2, 10]. Given an unsatisfiable instance, an MUS is an unsatisfiable subproblem such that every strict subproblem in it is satisfiable. In this paper we have studied MUS's in coloring, but use the graph theoretic name 'critical graph': a critical subgraph of an uncolourable graph is exactly an MUS of it. Gent and Walsh first noted that MUS's helped to explain ehi's [6]. They observed that ehi's typically had small and unique MUS's, while hard problems at phase transitions had MUS's which were larger and not unique. Our results are consistent with this, but suggest that on larger problems than they were able to examine, this behavior is likely to have changed. As in coloring, we would expect there to be a large jump in MUS size as small pathological graphs become vanishingly improbable.

There have been studies in SAT of how variables freeze to a single value. Parkes showed that at the satisfiability threshold, many variables have frozen although some are almost free [13]. He suggested that he would not expect a double phase transition in SAT. Monasson et al have used the frozen development in SAT to study the transition from P to NP [12]. Our studies also relate to this question, because problems will have good average case behavior when some polynomial check often helps to prove unsolvability. We suggest that in many NP complete problems, like graph coloring, there is no such cheap check and pathological behavior occurs at phase transitions in solvability as a result.

Chvatal and Szemeredi have shown that resolution proofs for overconstrained SAT problems are almost certainly long asymptotically [4]. Interestingly, this has not been shown for problems at a SAT phase transition, though it is likely that proofs in that region must be even longer. Extending this proof from resolution to any algorithm would prove NP $\neq$ co-NP, but it is likely that in SAT, coloring, and other NP problems, proofs of unsolvability must be long.

We conjecture that similar results will apply to other NP-complete problems in which phase transitions provide hard instances. Two obvious examples are satisfiability and general constraint satisfaction problems, but it would also be interesting to study problems such as the TSP, in which the analogue of critical graphs are not so obvious. A follow on in coloring would be to determine the minimum critical subgraph in each instance. Unfortunately this will likely prove difficult, possibly almost as hard as listing all critical graphs. This last would be the ultimate way to analyse the critical subgraphs of uncolorable graphs, but will inevitably be limited to small sizes. Further investigation is needed on how to compute all critical graphs [15].

8 Conclusions

We conclude that problems at the uncolorability phase transition are well out of reach of intelligent algorithms. Also, we conclude that there is little difference between graphs at the point where the first pair of nodes become frozen, and at the threshold in colorability. This suggests that the double phase transition in graph coloring is only an effect seen in small graphs: as the number of nodes increases the double phase transition converges to a single one. These results are likely to apply to any NP-complete problem where instances from phase transitions are hard for all known algorithms, and that our results help to explain this phenomenon.

Since there are not small and easily checkable subgraphs which can be used to confirm uncolorability quickly, we cannot hope to build more intelligent algorithms to avoid hard problems at the phase transition. For example, it is unlikely that learning algorithms would help significantly because the nogoods to be learnt must be inevitably be large.

More general lessons can be learnt from our study. An exciting application of our work is in understanding the difference between polynomial and exponential complexity. Many NP-complete problems have natural polynomial subclasses, for example 2-SAT within SAT. Since these subclasses are easily checkable, our results help to explain the sudden jump seen by Monasson et al from P-like behavior to NP-like behavior in the mixed 2+p-SAT model [12]. We expect that as the easily checked 2-SAT part of instances stops being responsible for unsolvability, large proofs will be necessary giving hard exponential behavior.

References

1. A.B. Baker. Intelligent backtracking on constraint satisfaction problems: Experimental and theoretical results. Technical report CIS-TR-95-08, Computer and Information Sciences, University of Oregon, 1995.
2. R.J. Bayardo Jr. and R.C. Schrag. Using CSP look-back techniques to solve exceptionally hard SAT instances. In *Principles and Practice of Constraint Programming – CP96*, pages 46–60. Springer, 1996.
3. P. Cheeseman, B. Kanefsky, and W.M. Taylor. Where the really hard problems are. In *Proceedings of the 12th IJCAI*, pages 331–337, 1991.
4. V. Chvatal and E. Szemeredi. Many hard examples for resolution. *Journal of the ACM*, 35(4):759–768, 1988.
5. I.P. Gent and T. Walsh. Easy problems are sometimes hard. *Artificial Intelligence*, pages 335–345, 1994.
6. I.P. Gent and T. Walsh. The satisfiability constraint gap. *Artificial Intelligence*, 81:59–80, 1996.
7. C.P. Gomes, B. Selman, and N. Crato. Heavy-tailed distributions in combinatorial search. In *Principles and Practice of Constraint Programming - CP-97*, pages 121–135. Springer, 1997. LNCS 1330.
8. T. Hogg and C. Williams. The hardest constraint problems: A double phase transition. *Artificial Intelligence*, 69:359–377, 1994.
9. A.D. Korshunov. The main properties of random graphs with a large number of vertices and edges. *Russian Math. Surveys*, pages 121–198, 1985.
10. D.L. Mammen and T. Hogg. A new look at the easy-hard-easy pattern of combinatorial search difficulty. *Journal of Artificial Intelligence Research*, 7:47–66, 1997.
11. D. Mitchell, B. Selman, and H. Levesque. Hard and easy distributions of SAT problems. In *Proceedings of AAAI-92*, pages 459–465, 1992.
12. R. Monasson, R. Zecchina, B. Selman, and L. Troyansky. 2+P-SAT: Relation of typical-case complexity to the nature of the phase transition:, 1999.
13. A. Parkes. Clustering at the phase transition. In *Proc. AAAI-97*, pages 340–345, 1997.
14. P. Prosser. An empirical study of phase transitions in binary constraint satisfaction problems. *Artificial Intelligence*, 81:127–154, 1996.
15. I. Ross. Finding all muses, 1997. Personal communication, isla@cs.strath.ac.uk.
16. B.M. Smith and M.E. Dyer. Locating the phase transition in binary constraint satisfaction problems. *Artificial Intelligence*, 81:155–181, 1996.
17. B.M. Smith and S.A. Grant. Sparse constraint graphs and exceptionally hard problems. In *Proceedings of IJCAI-95*, pages 646–651, 1995.
18. B.M. Smith and S.A. Grant. Modelling exceptionally hard constraint satisfaction problems. In *Principles and Practice of Constraint Programming - CP-97*, pages 182–195. Springer, 1997. LNCS 1330.
19. B. Vandegriend and J. Culberson. The gn,m phase transition is not hard for the Hamiltonian Cycle problem. *Journal of Artificial Intelligence Research*, 9:219–245, 1998.

9 Appendix

This appendix contains tabular versions of some graphs, as well as some data not contained in any form in the main body of the paper.

	First Frozen Size ($m^f + 1$)					Threshold Size (m^*)					Difference
n	Mean	Std	Mean/n	Min	Max	Mean	Std	Mean/n	Min	Max	Mean/n
25	75.04	8.91	3.00	54	96	87.68	6.23	3.51	69	100	0.51
50	177.56	13.57	3.55	140	197	193.04	9.74	3.86	169	215	0.31
75	287.18	15.13	3.83	240	309	304.34	10.16	4.06	275	325	0.23
100	398.56	12.44	3.99	355	414	414.20	10.16	4.14	383	433	0.15
125	512.82	13.31	4.10	442	530	526.46	10.48	4.21	505	550	0.11
150	623.92	8.84	4.16	603	641	635.80	12.53	4.24	607	671	0.08
175*	733.48	12.37	4.19	701	754	749.48	10.36	4.28	718	778	0.09

Table 5. Size of non-4-colorable graphs at first frozen(top) and at the threshold(bottom) for $k = 4$, 50 samples at each n (* except 175 for first frozen, where the sample is 27.)

	First Frozen Size ($m^f + 1$)					Threshold Size (m^*)					Difference
n	Mean	Std	Mean/n	Min	Max	Mean	Std	Mean/n	Min	Max	Mean/n
50	75.32	18.58	1.51	29	107	106.84	10.00	2.14	61	121	0.63
75	118.42	27.62	1.58	46	168	165.74	10.70	2.21	131	185	0.63
100	164.29	35.28	1.64	45	226	223.51	10.02	2.24	202	253	0.60
200	355.36	80.64	1.78	105	460	456.04	13.32	2.28	412	484	0.50
225	405.11	84.17	1.80	177	519	513.14	12.41	2.28	471	545	0.48
300	562.70	109.48	1.88	139	702	689.15	13.06	2.30	655	715	0.42

Table 6. Sizes of first frozen(top) and threshold non-colorable graphs, $k = 3$, 100 samples at each n. For $n = 50 \ldots 100$, the results come from the same set of graphs: this might increase the variance in the calculated difference, but should not affect the mean.

This article was processed using the LaTeX macro package with LLNCS style

Critical Subgraphs of First Frozen						
Small Cases			Remaining Cases			
n	#4-cliques	#≤ 12	#	Min Size	Max Size	Mean
50	83	3	14	25	53	41.93
75	80	4	16	13	109	61.75
100	81	3	16	13	149	102.44
200	78	3	19	234	353	296.95
225	73	2	25	248	412	346.36
300	78	1	21	407	595	510.57
350*	5	0	5	600	657	620.80

Critical Subgraphs of Threshold						
Small Cases			Remaining Cases			
n	#4-cliques	#≤ 12	#	Min Size	Max Size	Mean
50	9	3	88	15	76	47.36
50	11	2	87	15	80	48.09
75	2	0	98	16	131	78.04
75	2	0	98	16	151	79.97
100	0	0	100	27	161	114.27
100	0	0	100	45	174	114.73
125	3	0	97	78	211	162.03
150	3	0	97	36	276	202.90

Table 7. Size distributions of the critical subgraphs of the first frozen graphs, $k = 3$. 100 samples were taken at each n. In addition, two random critical subgraphs were computed for each of the threshold graphs for $n = 50 \ldots 100$, and thus the double listing for each of these entries. Due to long run times, only ten samples completed at $n = 350$ for first frozen. Notice that for larger n first-frozen graphs, there seem to be very few critical subgraphs that are between the 6-edge 4-clique and graphs with a few hundred edges.

Critical Sets of the First Frozen graphs					
n	Mean	Std	Mean/n	Min	Max
50	9.74	9.58	0.19	6	48
75	11.64	15.35	0.16	6	86
100	14.79	23.09	0.15	6	118
200	37.56	69.07	0.19	6	240
225	56.00	91.99	0.25	6	314
300	70.73	132.86	0.24	6	466

Critical Sets from Threshold graphs					
n	Mean	Std	Mean/n	Min	Max
50	16.80	11.62	0.34	1	49
75	26.96	18.68	0.36	1	83
100	31.85	21.12	0.32	1	85
125	41.50	29.92	0.33	3	160
150	51.18	34.63	0.34	5	140

Table 8. The critical sets, that is the number of edges that must occur in every critical subgraph of the first frozen and threshold non-3-colorable graphs.

	First Frozen Size (m^f+1)					Threshold Size (m^*)					Difference
n	Mean	Std	Mean/n	Min	Max	Mean	Std	Mean/n	Min	Max	Mean/n
50	117.08	8.31	2.34	105	141	138.62	14.58	2.77	114	175	0.43
75	170.16	8.23	2.27	149	194	192.38	11.45	2.57	174	224	0.30
100	228.82	8.26	2.29	214	249	251.64	11.28	2.52	230	280	0.23
125	290.91	7.10	2.33	273	304	309.69	7.87	2.48	290	328	0.15
150	349.42	7.41	2.33	335	363	368.10	10.82	2.45	344	401	0.12
175	408.26	7.04	2.33	393	425	424.30	8.68	2.42	408	449	0.09
200	466.02	8.06	2.33	449	481	484.30	10.06	2.42	464	508	0.09
225	525.02	8.98	2.33	503	546	542.90	7.49	2.41	522	561	0.08
250	583.54	9.77	2.33	556	603	600.10	12.61	2.40	573	635	0.07
275	644.38	7.89	2.34	625	660	660.52	10.21	2.40	636	693	0.06
300	703.38	12.20	2.34	660	730	721.32	11.93	2.40	689	747	0.06

Table 9. Empirical Measures of Frozen Development on Girth 4 $k=3$ Coloring Problems.

First Frozen Non-colorable					
N	Mean	Std	Ratio to Prev	Min	Max
50	2.24	3.11	—	0	12
75	5.66	5.66	2.53	0	24
100	15.04	15.89	2.66	0	77
125	42.16	35.45	2.80	1	177
150	77.84	47.19	1.85	9	178
175	197.64	161.24	2.54	12	714
200	299.68	229.42	1.52	27	1029
225	606.66	549.63	2.02	13	2795
250	1793.08	2622.69	2.96	22	16687
275	3496.66	4218.19	1.95	261	22971
300	8377.46	10755.89	2.40	311	55523

First Frozen Colorable					
N	Mean	Std	Ratio to Prev	Min	Max
50	11.50	3.22	—	6	19
75	13.62	3.64	1.18	6	24
100	16.94	5.40	1.24	5	31
125	22.32	9.70	1.32	8	64
150	32.24	18.33	1.44	13	126
175	53.48	37.44	1.66	19	166
200	64.30	48.77	1.20	20	198
225	138.18	153.13	2.15	28	961
250	389.60	588.32	2.82	25	2643
275	428.66	486.60	1.10	18	1876
300	1728.16	3689.39	4.03	35	23487

Threshold Non-colorable					
N	Mean	Std	Ratio to Prev	Min	Max
50	3.02	2.08	—	1	12
75	7.66	5.38	2.54	1	24
100	14.62	10.10	1.91	2	42
125	49.66	38.42	3.40	11	172
150	95.28	73.53	1.92	17	413
175	201.20	141.90	2.11	18	527
200	320.38	238.96	1.59	37	1169
225	704.28	555.17	2.20	20	2471
250	1793.58	2057.81	2.55	152	12292
275	3186.78	4418.67	1.78	542	31060
300	9381.28	21080.33	2.94	601	151296

Threshold Colorable					
N	Mean	Std	Ratio to Prev	Min	Max
50	8.68	2.99	—	3	16
75	12.98	5.48	1.50	5	28
100	20.98	8.97	1.62	8	45
125	42.48	28.89	2.02	10	138
150	59.52	44.90	1.40	13	252
175	132.06	113.98	2.22	20	533
200	175.52	157.86	1.33	19	598
225	396.46	470.56	2.26	19	2486
250	964.48	1245.71	2.43	25	6017
275	1314.36	1819.94	1.36	85	11964
300	5400.08	14984.26	4.11	74	105976

Table 10. Number of Backtrack Nodes to Solve Various instances in triangle-free graphs