Phase Transitions in the Properties of Random Graphs

Jeremy Frank[*] and **Charles U. Martel** [†]
Department of Computer Science
University of California at Davis
Davis, CA. 95616
August 25, 1995

Abstract

Problem instances of many $\mathcal{NP}$-Hard problems can be characterized by a parameter which describes how constrained the problem instance is. Based on this parameter instances can be partitioned into two natural classes: those which are mostly solvable and those which are mostly unsolvable. If classes are examined against a "constrainedness" axis, the transition from mostly solvable to mostly unsolvable occurs rapidly, and is called a *phase transitions*. Problem instances near this phase transition take more time, on the average, to solve using backtracking algorithms.

We present empirical results on the location of the phase transition for Hamiltonian cycle, complementing earlier theoretical results by Pósa. We relate these results to the phase transition for graph connectivity proved by Erdös and Renyi, and empirical results on the phase transition for graph biconnectivity. In particular, we present evidence that if a graph is biconnected then there is a high probability that it is also Hamiltonian. We use the phase transitions for easy graph problems to illuminate the phase for an $\mathcal{NP}$-Hard graph problem. Finally, we use these results to analyze the effectiveness of a fast probabilistic polynomial-time algorithm to find Hamiltonian cycles by Angluin and Valiant. We summarize with a set of strategies for solving Hamiltonian Cycle problems for differently constrained graphs.

[*]Correspondence Author. This author was supported by NSA URP MDA904-93-C-4085.

[†]This author was supported by NSF CCR-94-0365.

1 Introduction

$\mathcal{NP}$-Hard problems have captured the attention of researchers in theoretical computer science for the last 25 years. The tentative conclusions are that there is no no polynomial time algorithm guaranteed to solve all instances of $\mathcal{NP}$-Hard problems. These conclusions spurred research aimed at finding sub-classes of $\mathcal{NP}$-Hard problems which were provably easy to solve, i.e. had polynomial time solutions. Examples of such sub-classes are 2-SAT [Goe92] and Horn-SAT [Pap94]. Other researchers began searching for instances which were particularly hard to solve; these investigations were conducted by experimentalists rather than theorem-proving; these researchers measured the average time a program takes to solve problems. Some of this research concerns the use of *phase transitions*.

Results on phase transitions attack several interesting questions about $\mathcal{NP}$-Hard problems. Phase transitions have proven to be a good location for problem instances which are hard on the average for complete search algorithms to solve [CKT91]. There are interesting relationships between phase transitions in these problems and similar early results in mathematics and physical systems [KGTT94]. Much of this research has helped shed light on the phenomena in computer science. Most interesting of all is the potential of research on phase transitions to shed light on the structure of $\mathcal{NP}$-Hard problems. These insights may help us design better algorithms which take advantage of these properties.

A *phase transition* is a place where a system undergoes a sudden change of state. In $\mathcal{NP}$-Hard problems we are interested in the place where problem instances transit from the state of "most instances have solutions" to "most instances have no solution". In most cases there is a problem size independent *constraint parameter* which indicates how constrained the problem is. An example of a constraint parameter for the Hamiltonian cycle problem is the average degree of the graph; the fewer edges are in a graph, the fewer opportunities one has to find a HC. Phase transitions are typically shown by plotting the proportion of solvable problems with respect to the constraint parameter. A characteristic of phase transitions is that as the problem size increases, the curves become sharper; that is, when plotted against a constraint parameter, the transition occurs in a shorter interval of this parameter. In many cases a constraint parameter can be found such that all of the curves for different problem sizes intersect in the same place; this is called the *crossover point*. For Hamiltonian cycle, plotting the proportion of graphs with Hamiltonian cycles against the average degree does not result in a crossover point, but plotting against $\frac{E}{N \log N}$ does. In theory, these curves have a limiting form which is almost a step-like function, indicating a discrete shift from solvable to unsolvable problems.

The reason phase transitions are of interest to computer scientists is that there is a relationship between the phase transition for solvable instances of $\mathcal{NP}$-Hard problems and the average complexity of some algorithms to solve them. Measuring the average amount of time (or an equivalent complexity measurement such as number of states) required to solve many instances at different values of the constraint parameter provides evidence that the maximum average complexity is achieved at or near the phase transition. [CKT91] presented empirical results showing this relationship for Hamiltonian Cycle, Travelling Salesman, 3-SAT, and Graph Coloring. The transition for K-SAT has been extensively studied in [KGTT94] and [SK95]

Recent work has shown that the ever-tightening phase transition curves often have a single form when re-plotted against a function of the number of variables [KGTT94]. This phenomenon is called *scale invariance* and the so-called *scaling function* explains how the phase transition effect shapes up with increasing problem size. These functions are usually power laws, and have recently been applied to algorithm complexity data as well as solvability [SK95], [GW95a].

2 Phase Transitions in Random Graphs

In this paper we consider the phase transition for Hamiltonian Cycle [1]. We notice that a graph cannot have a Hamiltonian Cycle unless it is connected, or even *biconnected* [2]. We first define *random graphs*. We would like to select a graph over N nodes with E edges from all $\left(\begin{array}{c} \left(\begin{array}{c} N \\ 2 \end{array} \right) \\ E \end{array} \right)$ possible graphs. Since it is difficult to do this directly for even modest sized graphs, we give the following definition.

Definition 2.1 *Let G be a graph over N nodes and E edges. G is a* **random graph** *if G is formed by selecting E edges uniformly without replacement from the $\left(\begin{array}{c} N \\ 2 \end{array} \right)$ possible edges.*

There has been a good deal of theoretical work on phase transitions for various properties of random graphs. Erdös and Renyi proved that if $E = \frac{N \log N}{2} + cN$, for c a real constant, as $N \rightarrow \infty$ the probability of G being connected converges to $e^{-e^{-2c}}$. A corollary of this result is that if $E < \frac{N \log N}{2}$ the probability of G being connected goes to 0 and if $E > \frac{N \log N}{2}$ then the probability of G being connected goes to 1 [ER60]. Pósa used this result to show that if $E = cN \log N$, as $N \rightarrow \infty$ the probability of G containing a Hamiltonian cycle is 1 for c sufficiently large; Pósa uses $c = 30$ in his result [Pó76]. Angluin and Valiant later used this result to develop a fast algorithm to find Hamiltonian cycles. They show that their procedure finds Hamiltonian cycles with high probability in graphs with $E \geq cN \log N$ for c sufficiently large [AV]. While their result applies to both directed and undirected graphs, we restrict our discussion in this paper to undirected graphs only. Finally, Cheeseman et. al. [CKT91] present empirical results on the location of the phase transition for Hamiltonian Cycle and the location of the maximum average complexity of finding these cycles using a backtracking algorithm.

Section 3 discusses experiments to determine that the phase transition for biconnectivity occurs at $E = \frac{N \log N}{2}$. We also present a scale-invariance law for connectivity and for biconnectivity. Section 4 discusses experiments to find the phase transition for Hamiltonian cycle as well as a scaling law. We also relate the phase transition for Hamiltonian Cycle to that for biconnectivity. Since deciding biconnectivity of a graph is easy we hope that the phase transition for biconnectivity will help us understand the phase transition for Hamiltonian Cycle. Section 5 discusses the effectiveness of Angluin and Valiant's algorithm as well as some variants. Section 6 summarizes the results and presents some ongoing research.

3 The Phase Transitions for Graph Connectivity and Biconnectivity

The results of Erdös and Renyi describe the asymptotic behavior of the phase transition for random

[1] An undirected graph has a Hamiltonian Cycle if there exists a permutation of nodes such that node i in the permutation has an edge to node $i+1$ and node N of the permutation has an edge to node 1.

[2] An undirected graph is biconnected if there are two node-disjoint paths between every pair of nodes.

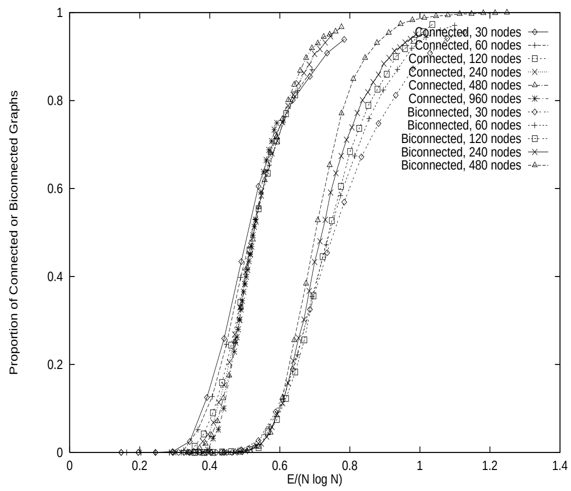

Figure 1: Graph Connectivity and Biconnectivity Phase Transition.

graphs but do not describe the behavior of graphs of moderate size. We proceeded to check these results for moderate sized graphs. Figure 1 shows our results. We plotted the proportion of 1000 graphs which were connected against $\frac{E}{N \log N}$ for various E; the left-most series of curves are those for connectivity. We note the sharpening behavior common to many phase transition results, and also the crossover point at $E \approx .55 N \log N$. This matches the earlier asymptotic results of Erdös and Renyi.

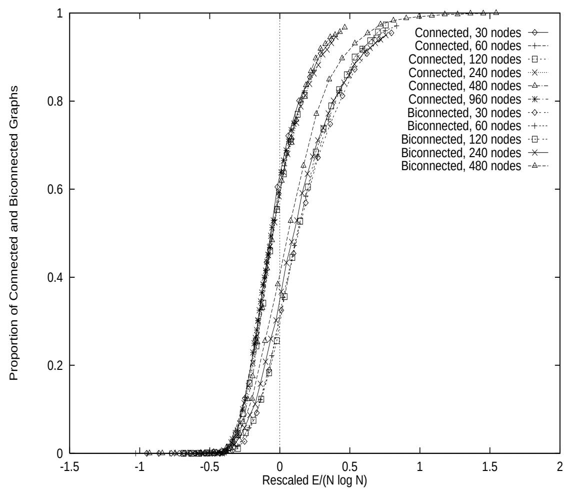

Figure 2: Scale Invariant Forms of Graph Connectivity and Biconnectivity Phase Transitions.

Our next task was to find a scale invariant form for the phase transition. The scale invariant law is a function for re-computing the x axis of a phase transition plot to eliminate the dependence of the curves on the size of the problem, leaving only the dependence on the constraint parameter. Taking our cues from previous research, we decided that the most likely functional form was $y = (N \log N)^{\frac{1}{\nu}}(x - \alpha_c)$, where α_c is the crossover point, N is the problem size and x is the value of the x-axis. Experimentation led to $\nu = 10$ as a good fit; the leftmost curve in figure 2 shows the replotted graphs.

We also decided to study the phase transition for graph biconnectivity. It is clear that all graphs with Hamiltonian cycles are biconnected, but there are biconnected graphs which do not have a Hamiltonian cycle. We plotted the proportion of 1000 graphs which were biconnected against $\frac{E}{N \log N}$ for various E. In figure 1 the rightmost series of curves show the phase transition for graph biconnectivity. We found the crossover point to be at about $\frac{E}{N \log N} \approx .68$, which is well separated from the crossover point for graph connectivity. We noted differences in the nature of the phase transitions for connectivity and biconnectivity. The crossover point for the connectivity phase occurs when about 75% of the graphs are connected, while the crossover point for biconnectivity occurs when about 35% of the graphs are biconnected. We also predict from the shape of the curves that the scale invariant law for biconnectivity would be different from that for connectivity. Experimentation led to $\nu = 8$ as a good fit; the rightmost curve in figure 2 shows the replotted graphs.

4 The Phase Transition For Hamiltonian Cycle

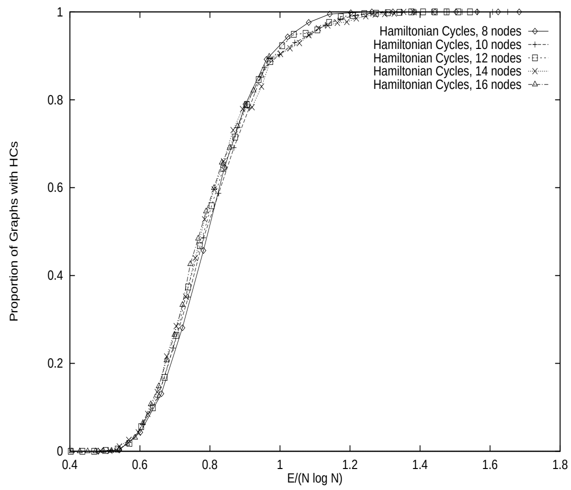

Figure 3: Hamiltonian Cycle Phase Transition.

We implemented a branch-and-bound algorithm to find Hamiltonian cycles. Because of the very poor performance of backtracking we were only able to test modest sized graphs. We plotted the proportion of 1000 graphs which contained Hamiltonian cycles against $\frac{E}{N \log N}$ for various E. Figure 3 shows this data. Unfortunately the graphs we tested were close enough together in size so that the sharpening phenomena is difficult to see. Examination of the curves indicated that the crossover point may occur at $\frac{E}{N \log N} \approx 0.7$. This is quite close to the crossover point for biconnectivity.

We measured the number of states the backtracking algorithm used to solve problems and measured the average for each value of E. We plotted the average rescaled so that the maximum complexity is 1. This data is shown in figure 4. We see results

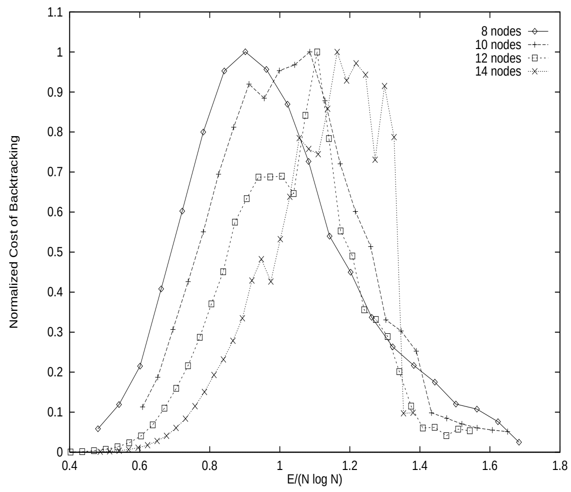

Figure 4: Normalized Complexity of Hamiltonian Cycle.

similar to those reported in [SK95]; the complexity peaks become narrower and narrower. We also notice that the maximum moves to the right with N; that is, for larger N the maximum cost occurs at a larger and larger value of $\frac{E}{N \log N}$. This is somewhat unusual, in that since we expect the curves to sharpen we also expect the peak to move to the left with N. Sharpening curves means that problems become mostly solvable for smaller values of $\frac{E}{N \log N}$, hence we expect the peak average complexity to occur at a smaller value of $\frac{E}{N \log N}$ also. Furthermore, there seems to be a double-peak behavior, in that a local maximum occurs at around $\frac{E}{N \log N} \approx .95$ with a second, higher maximum occurring at around $\frac{E}{N \log N} \approx 1.2$. We believe these results are due to the small problem sizes we had to test. We hoped to be able to find a scale invariance formula to match that for other problems. Our original conjecture was that the form would match that for the connectivity phase transition. Unfortunately, the nature of our data makes it unlikely that we could establish a good empirical estimate of the scaling function. We tested the function $y = (N \log N)^{\frac{1}{10}}(.7 - x)$ to replot our cost complexity data but the resulting curves did not match well.

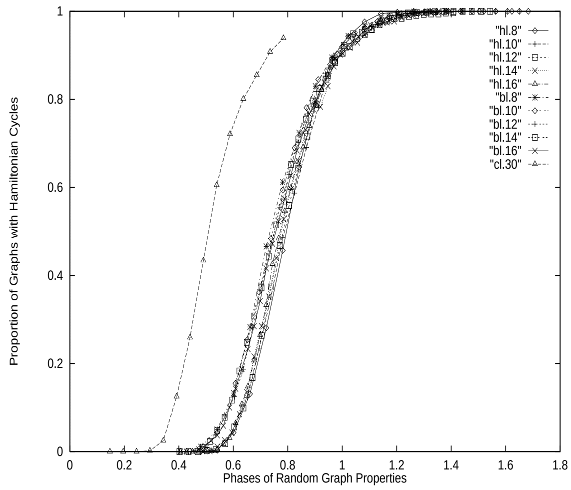

Figure 5: The Three Phase Transitions.

We notice that the phases for Hamiltonian cycle and connectivity are well-separated, but that the phase for biconnectivity lies very close to that for Hamiltonian cycle. Figure 5 shows the phase transitions for Hamiltonian cycle and biconnectivity and a reference curve for the phase transition for connectivity. We tested graphs with the same number of vertices for the sake of easy comparison. We see that for $.45 < \frac{E}{N \log N} < .65$ there are a very few graphs which are biconnected but do not have Hamiltonian cycles, and for $\frac{E}{N \log N} > .95$ it appears that there are almost no such graphs. In figure 6 we show the proportion of biconnected graphs which also have Hamiltonian Cycles. This gives us a picture of how useful it is to use biconnectivity as a test for the existence of Hamiltonian Cycles. For $.45 < \frac{E}{N \log N} < .7$ less than half of the biconnected graphs are Hamiltonian, and the backtracking algorithm is still comparatively inexpensive to run, so it is not efficient to use biconnectivity as a test. After $\frac{E}{N \log N} > .7$ most biconnected graphs are also Hamiltonian, so using biconnectivity may be an efficient pre-processing mechanism.

5 The Effectiveness of Fast Algorithms

Fast algorithms to solve SAT and K-SAT problems have been extremely successful at finding solutions to most large satisfiable formulae. While the best complete algorithms are still not capable of solving problems larger than $N = 1000$ variants of GSAT

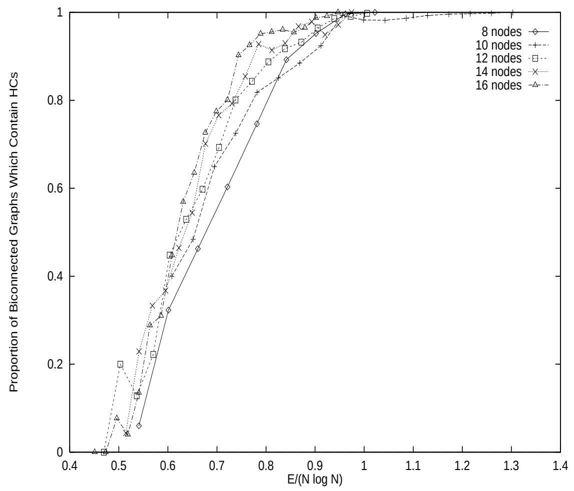

Figure 6: Proportions of Biconnected graphs which are Hamiltonian.

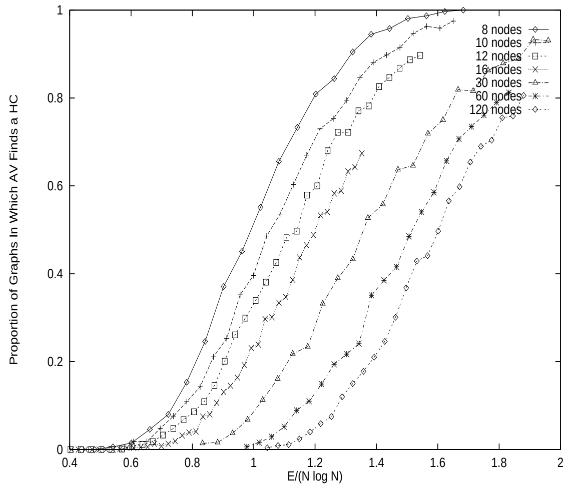

Figure 7: Satisfiability Performance of AV.

can quickly find the solution to satisfiable formulae of sizes near $N = 2000$ [BSC94]. Gent and Walsh report that many of these variants scale well with N [GW95b]. However, parameter setting for these algorithms can be difficult, and GSAT can still fail to solve problems which are satisfiable. Angluin and Valiant presented an $O(N(\log N)^2)$ time algorithm which finds a Hamiltonian cycle problem, if one exists, with high probability on random graphs with $cN \log N$ edges for sufficiently large c [AV]. The algorithm selects previously unselected edges at random from one end of a partial Hamiltonian cycle, called the *head*. If the edge selected "collides" with the path at some node i, the algorithm picks the edge from i to the node closest to the head, j, and eliminates this edge from the path. The algorithm then continues execution with j as the head. Figure

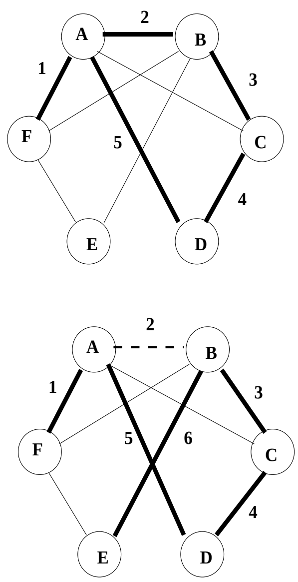

Figure 8: An Example of a Collision in Angluin and Valiant's Algorithm.

8 shows an example of this operation. At step 5 of the algorithm, D was the head and selected edge DA, causing a collision. In this case the node closest to the head from A is AB, so edge AB is eliminated from the graph and the algorithm proceeds with B as head, picking edge BE at step 6. Hence at step 6 the current path is $FADCBE$. Notice that the path "reverses" direction along the cycle formed by a collision, and that edges "deleted" from the path by collisions never will be considered again. For the rest of this paper we will refer to this algorithm as AV.

We tested the performance of AV by plotting the proportion of 1000 randomly generated graphs that the algorithm solved at different connectivities. We ran AV once per problem. We present data showing the effectiveness of this algorithm in figure 7. We note that these curves have a smaller slope and are shifted to the right. Hence Angluin and Valiant's algorithm is not as effective at solving HCs near the phase transition as the exact version. Angluin and Valiant's algorithm solves most of the problems by about $2.5N \log N$. We also note that the curves for large problems converge rather slowly, indicating that AV has more trouble with larger sized problems. Analyzing the results thus far, we attempt to estimate the eventual limiting form of the satisfiability curves. It appears that after $N = 120$ the curves will not change much more thereafter, but more data is necessary to show this conclusively.

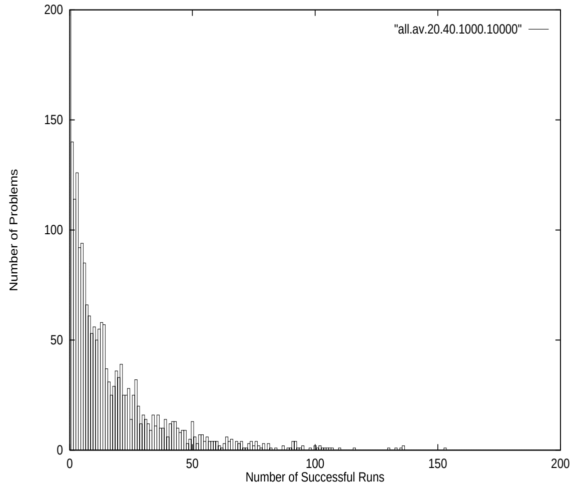

Figure 9: Satisfiability Performance of AV, $\frac{E}{N \log N} = .66$

Our data indicates that AV tends to perform poorly in the region $.5 < \frac{E}{N \log N} < 2$. How difficult is it to eliminate the "gap" in performance between AV and the estimated probability of existence of Hamiltonian cycles? One way of improving the performance of the algorithm is by repeated execution if the algorithm fails to find a Hamiltonian cycle. If we consider the monstrous cost of backtracking near the phase transition, executing an algorithm like AV even 10,000 times may be beneficial, especially given that AV executes in $O(N(\log N)^2)$ time. We executed AV 1000 times per problem for 10,000 problems of size 20 with $\frac{E}{N \log N} = .66, .84, 1.02$. We measured the probability that AV solved each problem instance, and plotted the number of problem instances which were solved with probability p. Figure 9 shows the distribution for $\frac{E}{N \log N} = .66$. Only 19% of the problem instances were solved at least once, which matches our expectation of the number of graphs with Hamiltonian cycles from the data in figure 3. However, almost all problem instances were solved less than 50 times out of 1000, indicating that AV had trouble solving these problems consistently. Figure 10 shows the distribution for $\frac{E}{N \log N} = .84$.

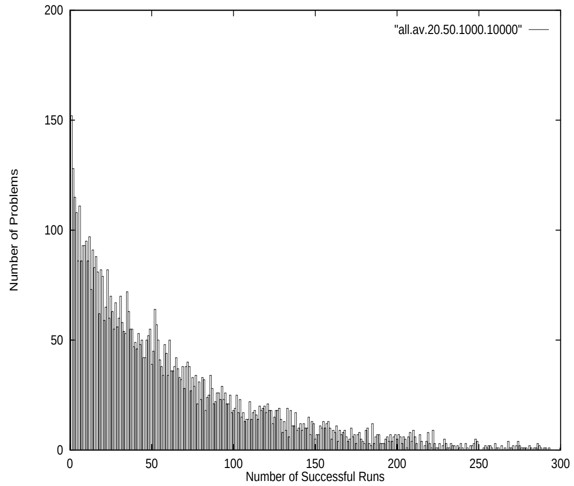

Figure 10: Satisfiability Performance of AV, $\frac{E}{N \log N} = .84$

We see that 65% of the problem instances were solved more than once, which again matches our expectation of the number of graphs with Hamiltonian cycles from the data in figure 3. The distribution has flattened out considerably, yet many problems are still difficult for AV to solve consistently. Figure 11 shows the distribution for $\frac{E}{N \log N} = .1.02$. We see that 90% of the problem instances were solved more than once, which again matches our expectation of

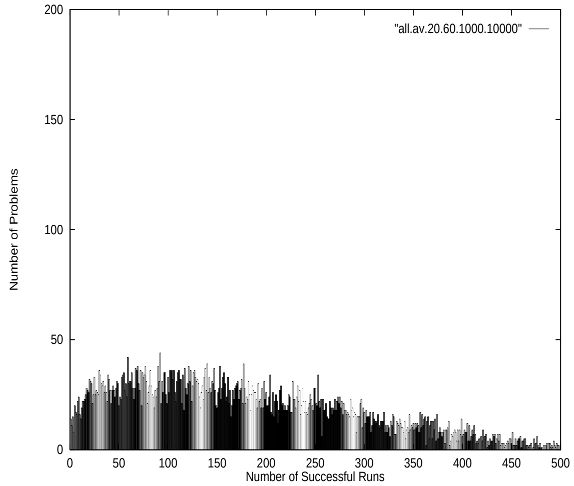

Figure 11: Satisfiability Performance of AV, $\frac{E}{N \log N} = .1.02$

the number of graphs with Hamiltonian cycles from the data in figure 3. The distribution has flattened out even more, and many problems are solvable by AV consistently.

Angluin and Valiant's algorithm has the greatest difficulty before backtracking becomes very expensive, and usually finds a Hamiltonian cycle if it exists when backtracking is too expensive, at least for the smaller sized graphs we have experimented with (i.e. < 60 nodes). We carried out experiments with graphs of 30 and 60 nodes using repeated executions of AV. With graphs of size 30, 1000 repetitions of AV seemed to solve the expected proportion of problem instances at all connectivities; with graphs of size 60, 4000 repetitions of AV was not enough to match the expected number of graphs with Hamiltonian cycles for $0.5 < \frac{E}{N \log N} < 1.0$. Figure 12 shows these re-

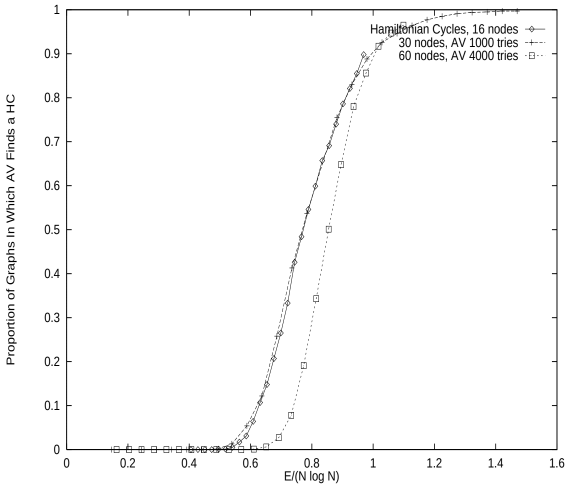

Figure 12: Satisfiability Performance of AV With Retries.

sults. More study is needed to find out exactly how AV scales with increasing N.

Empirical evidence suggests that when there are many edges, there are enough Hamiltonian cycles that making some wrong guesses does not harm the algorithm much, hence there is a high probability that AV can solve the problem. However, if there are few cycles and few edges, then there is little margin for error. These observations motivated us to try and improve the algorithm. Angluin and Valiant's original algorithm randomly selected the next edge to extend the current path. We implemented versions with two heuristics. One version selected the edge to the neighboring vertex with the fewest remaining untested edges, and the other selected the edge to the neighboring vertex with the most remaining untested edges. We believed that these heuristics could enhance the performance of Angluin and Valiant's algorithm by making intelligent choices about the next node on the path. Our belief was that the "least constrained first" heuristic would perform best since it brings the algorithm into regions of high density first; if the algorithm is forced to "backtrack" it will always do so into a region of high edge density. A major drawback to the "most constrained first" heuristic is the algorithm's method of handling "collisions"; it will probably exhaust the edges in the low edge density region and then find itself without any outbound edges, causing an incorrect failure. Surprisingly, there seemed to be little difference on small problems between the two heuristics, and neither differed significantly from the original version.

6 Conclusions and Future Work

We have presented empirical data complementing previous theoretical results on the location of the phase transitions for Hamiltonian cycle and Graph Connectivity. We also showed the scale invariant form of the connectivity phase and demonstrated that most biconnected graphs have a Hamiltonian cycle. Finally, we have examined the effectiveness of a fast probabilistic Hamiltonian cycle finding algorithm and begun analyzing it's effectiveness. These preliminary results suggest an appealing roadmap for solving Hamiltonian cycle problems. For graphs with low average degree, using the biconnectivity algorithm predicts whether or not most graphs have Hamiltonian cycles. For graphs with $.4 < \frac{E}{N \log N} < 1.0$ it appears that the growth of backtracking is just slow enough to make it worthwhile to use it; the performance of AV is poor enough to prevent it from being a good choice. However, when most graphs are likely to have Hamiltonian cycles, i.e. $\frac{E}{N \log N} > 1.0$, AV can find Hamiltonian cycles in most graphs with high probability given a sufficient number of repetitions.

With the experimental results on Hamiltonian cycles, it is now worthwhile to return to Pósa's results and attempt to prove mathematically that a Hamiltonian cycle exist with probability 1 when a random graph contains about $N \log N$ edges. Since there is such a close relationship to the phase transition for biconnectivity, it is also worth trying to prove that the phase transition for biconnectivity occurs in about the same place, perhaps using modifications of Erdös and Renyi's results.

We see opportunities for heuristics to further enhance Angluin and Valiant's algorithm. Since there are some problems which Angluin and Valiant has difficulty solving, even after repeated independent executions, using information from the previous runs might improve future runs. We think an approach similar to clause weights or using histories as in [SK95] might be appropriate. Another heuristic concentrates on avoiding "collisions", in which the node selected to extend the path is already on the path. Since a collision involves re-writing part of the path, it takes work. Also, a collision does not extend the path, but keeps it the same length. Finally, a collision removes an edge from consideration, thereby making the graph less dense and making it harder for the algorithm to find the Hamiltonian cycle. We feel that heuristics which avoided collisions might perform better, perhaps in combination with the heuristics mentioned above. However, it is important to realize that Angluin and Valiant's proof depends on the random selection of edges, and so their proof would not apply directly to the modified algorithm.

We advocate continued study of Angluin and Valiant's algorithm, particularly to see how it's performance scales with increasing problem size. We have not adequately demonstrated when it converges, and more work must be done to identify why AV performs so poorly in the region $.4 < \frac{E}{N \log N} < 1.0$. Options are to examine the graphs for which AV fails to find solutions often and analyze it's operation to see where the failures occur and why. This research will dovetail with the heuristics research proposed above.

Finally, we are considering studying the behavior of the phase transition for Hamiltonian Cycles in non-standard families of graphs. There are many results demonstrating that certain classes of graphs have Hamiltonian Cycles which are easy to find. We would like to study the phase transition for these classes of graphs, and hopefully other classes of graphs corresponding to real-world problems in order to further understand this phenomenon.

References

[AV] D Angluin and L. Valiant. Fast probabilistic algorithms for hamiltonian circuits and matchings.

[BSC94] H. Kautz B. Selman and B. Cohen. Noise strategies for improving local search. *Proceedings of 12th National Conference on Artificial Intelligence*, pages 337–343, 1994.

[CKT91] P. Cheeseman, B. Kanefski, and W. Taylor. Where the *really* hard problems are. *Proceedings of the 12th IJCAI*, pages 331–337, 1991.

[ER60] P. Erdös and A. Rényi. On the evolution of random graphs. *Publications of the Mathematical Institute of the Hungarian Academy of Sciences*, 5:569–573, 1960.

[Goe92] A. Goerdt. A threshold for unsatisfiability. *17th International Symposium of Mathematical Foundations of Computer Science*, pages 264–274, 1992.

[GW95a] I. Gent and T. Walsh. The tsp phase transition. Technical report, University of Strathclyde, 1995.

[GW95b] I. Gent and T. Walsh. Unsatisfied variables in local search. *AISB*, 1995.

[KGTT94] S. Kirkpatrick, G. Györgyi, N. Tishby, and L. Troyansky. The statistical mechanics of k-satisfaction. *Proceedings of the 7th Conference on Neural Information Processing Systems*, 1994.

[Ṕ76] L. Pósa. Hamiltonian circuits in random graphs. *Discrete Mathematics*, 14:359–364, 1976.

[Pap94] C. Papadimitrou. *Computational Complexity*. Addison Wesley, 1994.

[SK95] B. Selman and S. Kirkpatrick. Critical behavior in the computational cost of satisfiability testing. *AIJ Special Issue on Phase Transitions (To Appear)*, 1995.