Behavior of heuristics on large and hard satisfiability problems

John Ardelius and Erik Aurell*
Theoretical Physics, KTH—Royal Institute of Technology, AlbaNova University Center, SE-106 91 Stockholm, Sweden
(Received 1 February 2006; published 18 September 2006)

We study the behavior of a heuristic for solving random satisfiability problems by stochastic local search near the satisfiability threshold. The heuristic for average satisfiability (ASAT), is similar to the Focused Metropolis Search heuristic, and shares the property of being focused, i.e., only variables in unsatisfied clauses are updated in each step. It is significantly simpler than the benchmark WALKSAT heuristic. We show that ASAT solves instances as large as $N=10^6$ in linear time, on average, up to a ratio of 4.21 clauses per variable in random three-satisfiability. For K higher than 3, ASAT appears to solve instances of K-satisfiability up to the Montanari-Ricci-Tersenghi-Parisi full replica symmetry breaking (FSRB) threshold denoted $\alpha_s(K)$ in linear time.

DOI: 10.1103/PhysRevE.74.037702 PACS number(s): 02.70.−c, 05.40.−a, 64.60.Cn, 89.20.Ff

I. INTRODUCTION

Satisfiability (SAT) problems appear critically in many disciplines. Finding fast and reliable numerical methods for solving them is crucial in industrial applications, such as scheduling, or in verification.

The random K-satisfiability model (KSAT), where each condition or clause depends on the same number of variables K, and an instance is picked randomly with a given number of clauses per variable, has been of interest both to theoretical computer science and to statistical physics [1,2]. For $K \geqslant 3$KSAT belongs to the NP complete class of problems. While a deterministic algorithm will always find a solution if there is one, it presumably takes a long time to solve a KSAT instance in the worst case [3]. However, the typical behavior can be different. Indeed, since the beginning of the 1990s it has been known that the average running time of a deterministic algorithm depends on $\alpha=M/N$, the number of clauses (M) per variable (N) [4]. As α varies, a transition between a satisfiable and an unsatisfiable phase is observed at a threshold value α_c, and the transition becomes sharper as N increases [5]. Deterministic algorithms have longest average run times close to α_{cr} [4–6].

Stochastic search heuristics are not guaranteed to find a solution, if there is one, but may on the other hand greatly outperform a deterministic algorithm on a typical (solvable) instance. As α increases for given K and N, the typical run time of a given heuristic increases, eventually diverging, at the latest at α_c. The most interesting behavior, if it can be established, is if for some heuristic both the average run time grows only linearly in N for sufficiently small α, and also the distribution of run times per variable gets narrower as N increases. If so, the run time per variable is a self-averaging quantity. We denote here the greatest such α for some heuristic α_{lin} the linear-time transition for that heuristic.

A benchmark stochastic search is Papadimitriou's RANDOMWALKSAT [7]: in every step an unsatisfied clause is picked randomly, and then one random variable in that clause is flipped. For that algorithm, rate equations and direct simulations indicate that α_{lin} on 3SAT is approximately 2.7 [8,9]. Furthermore, for simple heuristics, such as straightforward guessing without backtracking, rate equations and direct simulations also show a nonzero α_{lin}, albeit smaller [10].

A limitation of RANDOMWALKSAT is that it does not distinguish which variable in a clause to flip, or if flipping one increases or decreases the number of unsatisfied clauses. The WALKSAT [11] algorithm mixes RANDOMWALKSAT moves with greedy steps, by default in equal proportion. WALKSAT is known to be quite powerful on SAT problems, but it was only shown quite recently to have a α_{lin} of 4.15 on 3SAT [12]. In contrast to RANDOMWALKSAT, rate equations have not been set up for WALKSAT: the interleaving of random and greedy moves, and the additional "freebie" move in the Selman-Kautz-Cohen heuristic, has made that complicated. Alava, Orponen, and Seitz showed that α_{lin} for WALKSAT could be pushed to or beyond 4.20 by optimizing over the proportion of random and greedy moves [13]. These authors further showed that two other algorithms Focused Metropolis Search (FMS) and Focused Record-to-Record Travel can be optimized to also have an apparent α_{lin} around 4.20. FMS in particular is quite simple: a variable in an unsatisfied clause is flipped if that decreases the number of unsatisfied clauses, and otherwise flipped or not flipped by a probability exponential in that number. FMS does not have the freebie move of WALKSAT, but is still of comparable efficiency.

In this paper we will introduce and study a heuristic *average SAT* (ASAT), which is arguably yet simpler than FMS. In ASAT a variable is flipped if this decreases the number of unsatisfied clauses, as in FMS, and then flipped with a constant probability if the number of unsatisfied clauses increases. ASAT is therefore sensitive to the widths of local minima, but not directly to the heights of the walls around local minima.

The relevance of these studies lies along the following lines. First, powerful search heuristics have an practical interest, and when they are simple one might hope for an analytical treatment along the lines of [8,9]. Second, from the theoretical side, it is of interest if α_{lin} for ASAT and other heuristics lies beyond 4.20 on random 3SAT, since that lies

*Also at SICS Swedish Institute of Computer Science AB, Kista, Sweden. Electronic mail: erik.aurell@physics.kth.se

beyond two natural candidates for upper bounds on α_{lin}, known, respectively, as α_d (which is around 3.92 for random 3SAT) and α_s (around 4.15). Such a high value is also quite close to the SAT-UNSAT transition α_c at 4.27.

The theoretical background of α_d and α_s can be briefly described as follows. Within the cavity method, it has been shown that for low enough α replica symmetry is unbroken, and the set of solutions is connected. In the interval $[\alpha_d, \alpha_{cr}]$ replica symmetry is broken, and the set breaks up into "clusters" [14,15], recently rigorously confirmed for large enough K [16,17]. The cavity method describes a one-step broken replica symmetry solution. In the interval $[\alpha_s, \alpha_c]$ it is unstable against a two-step broken replica symmetry perturbation [18]. The set of solutions is then presumably a hierarchical structure of clusters, as described by full replica symmetry breaking. For random 3SAT, α_s lies approximatively at 4.15.

A satisfiability problem is equivalent to the problem of finding a zero-energy ground state in a statistical mechanics model, where the "energy" is the number of unsatisfied clauses. In the UNSAT regime, where an instance is typically unsatisfiable, the ground state energy is typically larger than zero. In the SAT regime, clusters of solutions are local minima, which are also global minima. If these are accompanied by a much larger number of clusters with nonzero energy, such clusters could act as traps to local search heuristics. The number of clusters of local minima at given energy was computed by the cavity method for random 3SAT in [14,15], and for higher K in [19], and does increase with energy.

The results of this paper and of [13] indicate that α_{lin} for optimized algorithms is substantially larger than α_s for 3SAT. We show that the run time of ASAT is self-averaging at $\alpha=4.21$ on 3SAT up to instances of one million variables, while it is not self-averaging at $\alpha=4.25$. We present here data that at $\alpha_s(K)$, as computed recently in [19], run times of ASAT seem self-averaging up to $K=7$. The time course of a solution is another quantity of interest. Below α_{lin}, ASAT solves an instance in linear time, similarly to RANDOMWALKSAT below its α_{lin}. Above α_{lin}, ASAT typically solves an instance by a slow process, "sinking" through several plateaus. We show results from one such run, and we note that it appears to be different from the "solution by fluctuations" proposed for RANDOMWALKSAT above its α_{lin} [8]. To optimize ASAT we introduce a reheating procedure.

II. THE ASAT HEURISTIC

ASAT is a focused heuristic, like RANDOMWALKSAT, WALKSAT, and FMS, meaning it focuses on the unsatisfied clauses at any given time, and makes trial moves only to neighboring states by flipping a variable that appears in at least one unsatisfied clause. Variables that only appear in satisfied clauses are never flipped. For any trial move, ASAT computes only if that move will increase or decrease the energy (number of unsatisfied clauses). A move that increases the energy will be accepted with fixed probability, independent of how much the energy is changed, while a move that decreases the energy will be accepted always.

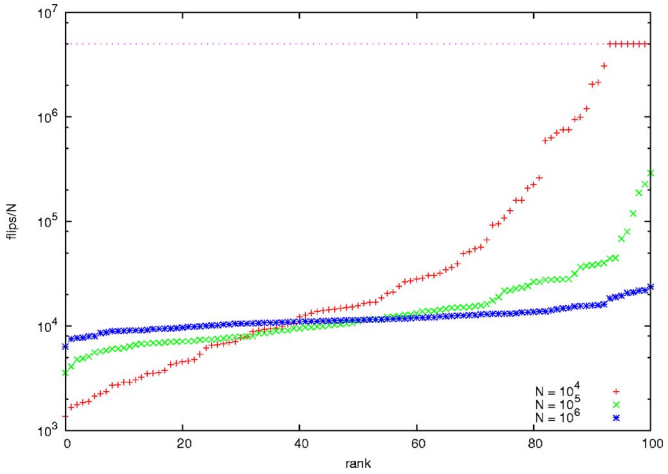

FIG. 1. (Color online) Ranked logarithmic run times per variable of ASAT, $p=0.21$, at $\alpha=4.21$ and values of N from 10^4 to 10^6 on 3SAT. Note pivoting of the distributions, as in [12]. Note that all runs were made with a cutoff of $5\times10^6 N$ flips. Out of 100, all instances at N equal to 10^5 and 10^6 are solved within this time; most instances at $N=10^6$ taking close to 10^{10} flips. For the smallest size, $N=10^4$, the spread is larger, and about 10% of the instances are not solved in 5×10^{10} flips, although the median is but a little more than 10^8 flips.

The ASAT algorithm is therefore characterized by the single parameter p, which plays an analogous role to the proportion of random and greedy moves in WALKSAT (a parameter also called p), and the noise parameter η of FMS. Optimization of p in ASAT is discussed below in Sec. III. Figure 1 shows a rank ordered plot of the run times for different system sizes $N=10^4, 10^5, 10^6$ at $\alpha=4.21$. This and analogous data for several lower values of α (data not shown) indicate that the run time of ASAT is self-averaging at least this far.

On the other hand, at $\alpha=4.25$, the conjectured end point of the Survey Induced Decimation (SID) algorithm [14,15], ASAT is not self-averaging (data not shown).

The solution process can be characterized by the fraction of unsatisfied clauses as a function of the number of flips. Following [8–10] it is convenient to introduce a "time" as (flips)/N. Figure 2 shows the time course of a solution process at $\alpha=4.22$. One can clearly see three regimes, one fast, one intermediate, and one quite slow. The fast regime, up to time about ten thousand, is presumably analogous to the Poissonian regime in RANDOMWALKSAT as studied by [8,9]. The intermediate and slow regimes have, as far as we know, not been shown on these problems previously. We note that dynamics appears self-averaging in both the fast and the intermediate regimes. The slow regime proceeds by plateaus (long waiting periods). While this qualitative behavior repeats itself from run to run, the position and lengths of the plateaus do not.

Finally, we have investigated ASAT for K larger than 3, albeit in less detail. While computational determination of the threshold gets harder at higher values of K, one may look for evidence that some given α is comfortably below α_{lin}. In Fig. 3 we have looked at $\alpha_s(K)$, whose values were recently

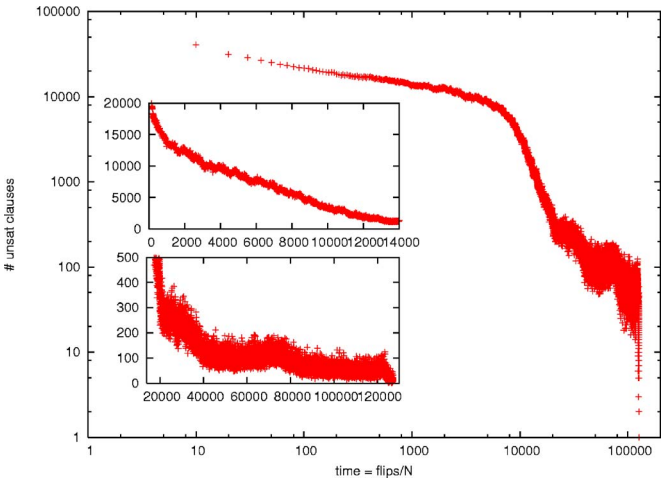

FIG. 2. (Color online) Time course of solution of one instance by ASAT at $\alpha=4.22$ and $N=10^6$. Note that ASAT solves this instance, but only after 1.3×10^5 time steps, i.e., 1.3×10^{11} flips. The main plot, in logarithmic coordinates, shows that the solution proceeds in three stages. First, there is decay on a time scale up to about 10^3. This process slows down, and is overtaken by another process which last up to time about 2×10^4. Finally, there is a very slow decrease to the solution. Top inset shows a blowup of the second stage in linear coordinates. Bottom inset shows the final decrease, which proceeds via plateaus where the energy is approximately constant. In this run three plateaus can be discerned, with approximately 200, 50, and 15 unsatisfied variables, respectively.

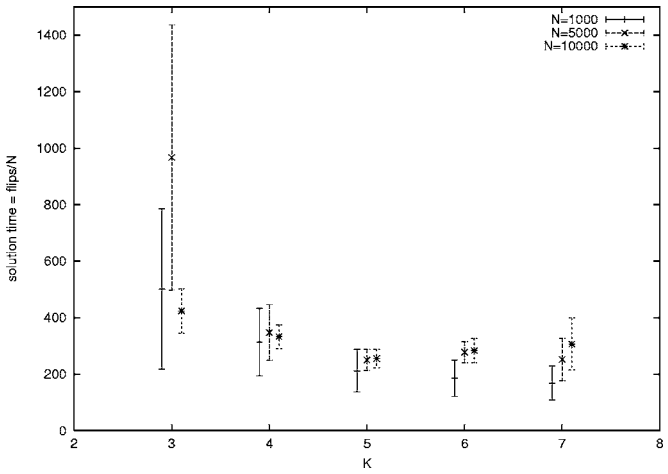

FIG. 3. Median and quartile run times per variable at $\alpha=\alpha_s(K)$ (values taken from [19]). The figure indicates that ASAT solves these instances with about equal computational cost per variable, for all K. The parameter p was found by the procedure ASAT-HEAT (see main text) on each value of K separately. The values were 0.21, 0.118, 0.068, 0.045, and 0.032 for K from 3 to 7, respectively.

given as 4.15, 9.08, 17.8, 33.6, and 62.5 for K from 3 to 7, respectively [19]. The results are not entirely conclusive, but tend to support that α_{lin} is greater than α_s.

III. PARAMETER OPTIMIZATION

In this section we describe a method to optimize the value of the noise parameter p, a method we call simulated heating of ASAT (ASAT-HEAT).

The idea is that there is a tradeoff to be made between an algorithm getting out of local minima and efficiently exploring the bottom of a local minimum. Hence, the premise is that solutions are found at the bottom of *some* local minima, which are not otherwise distinguished. Recently, we became aware that a related idea, "optimization at the ergodic edge," has recently been considered by Boettcher and Frank [20], and also used to optimize the Record-to-Record-Travel algorithm, by Jia, Moore, and Selman [21].

In the context of ASAT we look for the value p such that the algorithm does not get stuck, while still exploring the bottom of mimina where it finds itself. That is done by an interleaved process, where the algorithm alternatively runs with some nonzero p (to explore phase space, and get out of minima), and alternatively freezes at zero p (to find the bottom of the mininum it is moving in). The value of p_{cr} found by ASAT-HEAT decreases with increasing K; the values for K from 3 to 7 are given in the caption to Fig. 3

IV. DISCUSSION

We have in this work presented a heuristic for satisfiability problems called ASAT. We have shown that ASAT has typical run time linear in N up to $\alpha=4.21$ on 3SAT, up to the largest instances that can conveniently be studied ($N\sim10^6$). This means, that to the best of our estimate, the linearity threshold for ASAT, α_{lin}, is larger than that for the clustering transition ($\alpha_d=3.92$) and also the FRSB threshold ($\alpha_s=4.15$). We have studied ASAT at larger K, and showed that α_{lin} is likely to be larger than α_s there also. A parameter optimization technique ASAT-HEAT was introduced. This allows for a determination of an optimal parameter value of the algorithm, and can be considered an alternative to the extensive simulations at many values of N, α, and one algorithm parameter used in [13].

While physical intuition suggests that local heuristics will have difficulties where many metastable states appear this does not seem to be the case. RANDOMWALKSAT and very simple heuristics have difficulties far below α_d, while ASAT and other heuristics seem to work linearly beyond α_s. Let us therefore end by stating the differences between stochastic local search heuristics to find satisfying assignments in random KSAT and a physical process of random walk in a corresponding energy landscape. First, RANDOMWALKSAT, WALKSAT, FMS, and ASAT are focused: these algorithms correspond to nonequilibrium dynamics without detailed balance [9]. Conservation of cluster structure under nonequilibrium perturbations is a delicate point in some spin glass models [22,23]. Second, while FMS is similar to a random walk in the energy landscape, in the sense that the dynamics directly depends on the local energy, WALKSAT and ASAT are not. In WALKSAT with the Selman-Kautz-Cohen heuristic, decisions are based on a change in BREAKCLAUSE which is not

the same as an energy change, while in ASAT decisions are based on whether the energy increases or decreases at all. We do not know if this is an essential difference. Therefore, finally, while numerical simulations cannot rule out that, e.g., ASAT will run into trouble beyond α_s on instances larger than the ones we have studied, we are not sure if it necessarily has to. Theoretical predictions on for what N one would expect to see nonlinear behavior at what α would be most helpful to guide numerical experiments on this issue.

ACKNOWLEDGMENTS

This work was supported by the Swedish Research Council (E.A). We thank Supriya Krishnamurthy for numerous discussions and a close reading of the manuscript. We thank Mikko Alava, Jean-Philippe Bouchaud, Scott Kirkpatrick, Pekka Orponen, Giorgio Parisi, Sakari Seitz, Guilhem Semerjian, and Riccardo Zecchina for discussions and critical remarks.

[1] A. K. Hartmann and M. Weigt, *Phase Transitions in Combinatorial Optimization Problems* (Wiley-VCH, 2001).
[2] *NP-Hardness and Phase Transitions*, O. Dubois, R. Monasson, B. Delman, and R. Zecchina, eds., Theor. Comput. Sci. 265 (2001).
[3] Stephen Cook, www.claymath.org/prizeproblems/pvsnp.htm
[4] D. Mitchell, B. Selman, and H. Levesque, in Proceedings of the Tenth National Conference on Artificial Intelligence AAAI-92 (AAAI Press, San Jose, CA, July 1992), pp. 456–465.
[5] S. Kirkpatrick and B. Selman, Science **264**, 1297 (1994).
[6] S. Cocco and R. Monasson, Phys. Rev. Lett. **86**, 1654 (2001).
[7] C. H. Papadimitriou, in *Proceedings of the 32nd Annual Symposium on Foundations of Computer Science* (IEEE Computer Society Press, New York, 1991), pp. 163–169.
[8] W. Barthel, A. K. Hartmann, and M. Weight, Phys. Rev. E **67**, 066104 (2003).
[9] G. Semerjian and R. Monasson, Phys. Rev. E **67**, 066103 (2003).
[10] C. Deroulers and R. Monasson, *Advances in Neural Information Processing Systems 17*, edited by Lawrence K. Saul, Yair Weiss, and Léon Bottou (American Mathematical Society, Providence, Rhode Island, 1996), pp 521–532.
[11] B. Selman, H. Kautz, and B. Cohen, in *Cliques, Coloring, and Satisfiability*, DIMAC Series in Discrete Mathematics and Theoretical Computer Science (American Mathematical Society, Providence, Rhode Island, 1996), pp 521–532.
[12] E. Aurell, U. Gordon, and S. Kirkpatrick, in Eighteenth Annual Conference on Neural Information Processing Systems (NIPS), 2004.
[13] S. Seitz, M. Alva, and P. Orponen, J. Stat. Mech.: Theory Exp. (2005) p. 06006.
[14] M. Mézard and R. Zecchina, Phys. Rev. E **66**, 056126 (2002).
[15] M. Mézard, G. Parisi, and R. Zecchina, Science **297**, 812 (2002).
[16] M. Mézard, T. Mora, and R. Zecchina, Phys. Rev. Lett. **94**, 197205 (2005).
[17] D. Achlioptas, A. Naor, and Y. Peres, Nature (London) **435**, 759 (2005).
[18] A. Montanari, G. Parisi, and F. Ricci-Tersinghi, J. Phys. A **37**, 2073 (2004).
[19] S. Mertens, M. Mézard, and R. Zecchina, Random Struct. Algorithms **28**, 340 (2006).
[20] S. Boettcher and M. Frank, Physica A **367**, 220 (2006).
[21] H. Jia, C. Moore, and B. Selman, in *Theory and Applications of Satisfiability Testing, 7th International Conference, SAT 2004*, Lecture Notes in Computer Science, Vol. 3542 (Springer, Berlin, 2005), p. 199.
[22] A. Crisanti and H. Sompolinsky, Phys. Rev. A **36**, 4922 (1987).
[23] L. Cugliandolo, J. Kurchan, P. L. Doussal, and L. Peliti, Phys. Rev. Lett. **78**, 350 (1997).