Restarts and exponential acceleration of the Davis–Putnam–Loveland–Logemann algorithm: a large deviation analysis of the generalized unit clause heuristic for random 3-SAT.

Simona Cocco [1] and Rémi Monasson [2,3]
[1] *CNRS-Laboratoire de Dynamique des Fluides Complexes, 3 rue de l'Université, 67000 Strasbourg, France.*
[2] *CNRS-Laboratoire de Physique Théorique de l'ENS, 24 rue Lhomond, 75005 Paris, France.*
[3] *CNRS-Laboratoire de Physique Théorique, 3 rue de l'Université, 67000 Strasbourg, France.*

Abstract. An analysis of the hardness of resolution of random 3-SAT instances using the Davis-Putnam-Loveland-Logemann (DPLL) algorithm slightly below threshold is presented. While finding a solution for such instances demands exponential effort with high probability, we show that an exponentially small fraction of resolutions require a computation scaling linearly in the size of the instance only. We compute analytically this exponentially small probability of easy resolutions from a large deviation analysis of DPLL with the Generalized Unit Clause search heuristic, and show that the corresponding exponent is smaller (in absolute value) than the growth exponent of the typical resolution time. Our study therefore gives some quantitative basis to heuristic restart solving procedures, and suggests a natural cut-off cost (the size of the instance) for the restart.

Keywords: restart, satisfiability, DPLL, large deviations.

1. Introduction.

Being a NP-complete problem, 3-SAT is not thought to be solvable in an efficient way, *i.e.* in time growing at most polynomially with N. In practice, one therefore resorts to methods that need, *a priori*, exponentially large computational resources. One of these algorithms is the ubiquitous Davis–Putnam–Loveland–Logemann (DPLL) solving procedure (Davis, Logemann and Loveland, 1962; Gu, Purdom, Franco and Wah, 1997). DPLL is a complete search algorithm based on backtracking. The sequence of assignments of variables made by DPLL in the course of instance solving can be represented as a search tree, whose size Q (number of nodes) is a convenient measure of the resolution time. Some examples of search trees are presented in Figure 1.

In the past few years, many experimental and theoretical progresses have been made on the probabilistic analysis of 3-SAT (Hogg, Huberman and Williams, 1996; Gent, van Maaren and Walsh, 2000). Distributions of random instances controlled by few parameters are particularly useful in shedding light on the onset of complexity. An example that has attracted a lot of attention is random 3-SAT: the three literals in a clause are randomly chosen variables, or their negations, with equal probabilities, among a set of N Boolean variables; clauses are drawn independently of each other. Experiments (Hogg, Huberman and Williams, 1996; Mitchell, Selman and Levesque, 1992; Crawford and Auton, 1996; Kirkpatrick and Selman, 1994) and theory (Friedgut, 1999; Dubois, Boufkhad and Mandler, 2000; Kaporis, Kirousis and Lalas, 2002; Dubois et al., 2001) indicate that clauses can almost surely always (respectively never) be simultaneously satisfied if α is smaller (resp. larger) than a critical threshold $\alpha_C \simeq 4.3$ as soon as M, N go to infinity at a fixed ratio α. This threshold phenomenon, similar to phase transitions in condensed matter physics (Monasson et al., 1999), is accompanied by a drastic increase in hardness (Hogg, Huberman and Williams, 1996; Mitchell, Selman and Levesque, 1992; Crawford and Auton, 1996).

The emerging pattern of complexity is as follows. At small ratios $\alpha < \alpha_L$, where α_L depends on the heuristic used by DPLL, instances are almost surely satisfiable (sat) (Franco, 2001; Achlioptas, 2001b). Finding a solution requires a tree whose size Q scales only linearly with the size N, and almost no backtracking is present (Figure 1A). Above the critical ratio α_C, instances are almost surely unsatisfiable (unsat) and proofs of refutation are obtained through massive backtracking. Chvátal and Szémeredi proved that the hardness scales exponentially with the instance size, $Q = 2^{N\omega}$ with $\omega > 0$ (Chvàtal and Szmeredi, 1988). This result was sharpened by relating ω to α, $\omega = O(1/\alpha)$ at large ratios (Beame et al., 1998). The intermediate range of ratios, $\alpha_L < \alpha < \alpha_C$, hereafter referred to as upper sat phase is of particular interest. There, instances are almost surely sat, but experiments and heuristic arguments indicate that resolution requires an exponentially large computational effort ($\omega > 0$, corresponding to the search tree sketched in Figure 1B) with high probability (Coarfa et al., 2000; Cocco and Monasson, 2001). This exponential behaviour was rigorously established to hold for ratios $\alpha < \alpha_C$ *i.e.* below the threshold and, with an additional hypothesis presented in Section 2, in the whole upper sat phase (Achlioptas, Beame and Molloy, 2001c). In addition, a quantitatively accurate but non-rigorous calculation of ω as a function of α in both upper sat and unsat phases was recently made available (Cocco and Monasson, 2001).

In this paper, we study in more detail the distribution of resolutions complexities of randomly drawn instances with ratios $\alpha_L < \alpha < \alpha_C$.

Using numerical experiments and analytical calculations, we show that, though complexity Q almost surely grows as $2^{N\omega}$, there is a finite but exponentially small probability $2^{-N\zeta}$ that resolution never requires backtracking and Q is bounded from above by $O(N)$ only. In other words, while finding solutions to these sat instances is almost always exponentially hard, it is very rarely easy (polynomial time). Taking advantage of the fact that ζ is smaller than ω, we show how systematic restarts of the heuristic may decrease substantially the overall search cost. Our study therefore gives some theoretical basis to incomplete restart techniques known to be efficient to solve satisfiable instances (Dubois et al., 1993; Gomes et al., 2000), and suggests a natural cut-off cost for the restart (Cocco and Monasson, 2002).

We first start by recalling the standard framework for studying resolutions taking place with high probability, which is also helpful in understand rare resolutions (section 2). Numerical experiments are presented in section 3. Section 4 is devoted to the analytical calculation of ζ, and we present some conclusions in section 5. Throughout the paper, we have tried to highlight the different status of the statements and results presented: rigorous, expected to be exact but lacking proof, approximate, or experimental. We hope this effort will benefit to the reader, and make our work more accessible.

2. Resolution trajectories.

2.1. The high probability scenario.

In this section, we briefly recall the main features of the resolution by DPLL of satisfiable instances of size N, occurring with large probability as $N \to \infty$. This framework is largely based on pioneering works by Chao and Franco (1986; 1990). A recent review of these and subsequent works can be found in (Achlioptas, 2001b).

The action of DPLL on an instance of 3-SAT causes the changes of the overall numbers of variables and clauses, and thus of the ratio α. Furthermore, DPLL reduces some 3-clauses to 2-clauses, and 2-clauses to 1-clauses. As the latter are eliminated through unit propagation, a mixed distribution (Chao and Franco, 1986), referred to as 2+p-SAT with p the fraction of 3-clauses, can be used to model what remains of the input instance at a node of the search tree. Using experiments and methods from statistical mechanics (Monasson et al., 1999), the threshold line $\alpha_C(p)$, separating sat from unsat phases, may be estimated with the results shown in Figure 2. For $p \leq p_T = 2/5$, *i.e.* left to point T, the threshold line is given by $\alpha_C(p) = 1/(1-p)$, and saturates

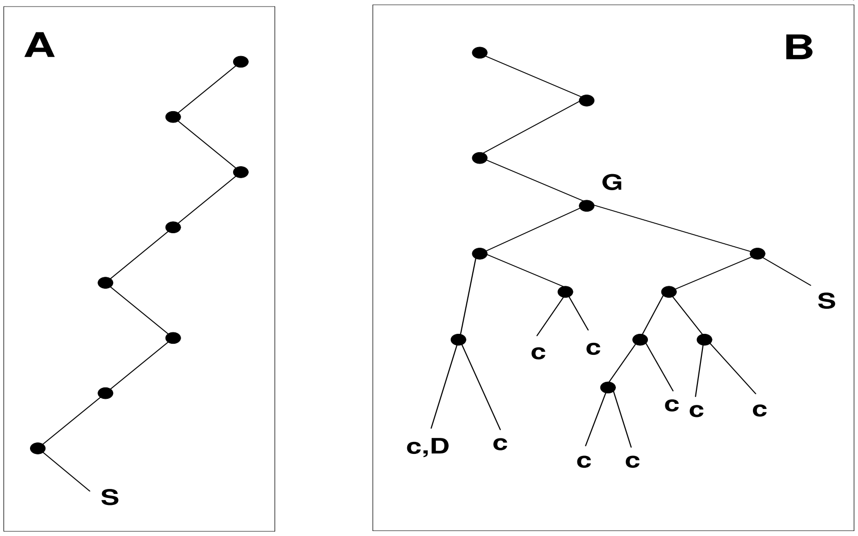

Figure 1. Types of search trees generated by the DPLL solving procedure on satisfiable instances. **A.** *lower sat phase,* $\alpha < \alpha_L$: the algorithm finds easily a solution with almost no backtracking. **B.** *upper sat phase,* $\alpha_L < \alpha < \alpha_C$: many contradictions (c) arise before reaching a solution, and backtracking enters massively into play. Junction G is the highest node in the tree reached back by DPLL. D denotes the first contradiction detected by DPLL, located at the leaf of the first descent in the tree.

the upper bound for the satisfaction of 2-clauses (Achlioptas et al., 2001a). Above p_T, no exact expression for $\alpha_C(p)$ is known.

The phase diagram of 2+p-SAT is the natural space in which the DPLL dynamic takes place. An input 3-SAT instance with ratio α shows up on the right vertical boundary of Figure 2 as a point of coordinates $(p = 1, \alpha)$. Under the action of DPLL, the representative point moves aside from the 3-SAT axis and follows a trajectory which depends on the splitting heuristic implemented in DPLL. We consider here the so-called Generalized Unit-Clause (GUC) heuristic proposed by Chao and Franco(1986; 1990). Literals are picked up randomly among one of the shortest available clauses. This heuristic does not induce any bias nor correlation in the instances distribution (Chao and Franco, 1986). Such a statistical "invariance" is required to ensure that the dynamical evolution generated by DPLL remains confined to the phase diagram of Figure 2.

Frieze and Suen (1996) completed the rigorous analysis of resolutions corresponding to initial ratios $\alpha < \alpha_L \simeq 3.003$. Their analysis consists in monitoring the evolution of the densities (numbers divided by N) c_2 and c_3 of 2- and 3-clauses respectively as more and more variables are assigned by DPLL and the search tree, or branch, of Figure 1A develops. Both densities become highly concentrated around the averages as the size N goes to infinity. Calling t the fraction of assigned variables, $c_2(t)$ and $c_3(t)$ obeys a set of coupled ordinary differential equations (ODE),

$$\begin{aligned} \frac{dc_3(t)}{dt} &= -\frac{3}{1-t} c_3(t) - \rho_3(t) \\ \frac{dc_2(t)}{dt} &= \frac{3}{2(1-t)} c_3(t) - \frac{2}{1-t} c_2(t) - \rho_2(t) \quad , \end{aligned} \tag{1}$$

where $\rho_2(t), \rho_3(t)$ are the probabilities that the split is made from a 2-, 3-clause respectively. For GUC and an initial ratio $\alpha_0 > 2/3$, $\rho_2(t) = 1 - c_2(t)/(1-t), \rho_3(t) = 0$.

Resolution of the ODEs (1) with initial conditions $c_2(0) = 0, c_3(0) = \alpha_0$, and change of variables

$$\alpha(t) = \frac{c_2(t) + c_3(t)}{1-t} \quad , \quad p(t) = \frac{c_3(t)}{c_2(t) + c_3(t)} \quad . \tag{2}$$

permit us to obtain the branch trajectories in the phase diagram of Figure 2,

$$\alpha(t) = \frac{\alpha_0}{4}(1-t)^2 + \frac{3\alpha_0}{4} + \ln(1-t) \quad , \quad p(t) = \frac{\alpha_0 (1-t)^2}{\alpha(t)} \quad . \tag{3}$$

Results are shown for the GUC heuristics and starting ratios $\alpha_0 = 2$ and 2.8 in Figure 2. The trajectory, indicated with a dashed line, first heads to the left and then reverses to the right until reaching a point on the 3-SAT axis at a small ratio. Further action of DPLL leads to a rapid elimination of the remaining clauses and the trajectory ends up at the right lower corner S, where a solution is found. Note that for initial ratios $\alpha_0 < 2/3$, only the second part of the trajectory restricted to the $p = 1$ axis subsists. The average length Q of the branch scales linearly with N, with a multiplicative factor $\gamma(\alpha_0) = Q/N$ that can be computed exactly (Cocco and Monasson, 2001b).

The boundary α_L of this easy sat region can be defined as the largest initial ratio α_0 such that the branch trajectory $p(t), \alpha(t)$ issued from α_0 never leaves the sat phase in the course of DPLL resolution. Indeed, as α_0 increases up to α_L, the trajectory gets closer and closer to the threshold line $\alpha_C(p)$. Finally, at $\alpha_L \simeq 3.003$, the trajectory touches the threshold curve tangentially at point T.

The concept of trajectory helps to understand how resolution takes place in the upper sat phase, that is for ratios α_0 ranging from α_L to α_C.

The branch trajectory, started from the point $(p = 1, \alpha_0)$ corresponding to the initial 3-SAT instance, hits the critical line $\alpha_c(p)$ at some point G with coordinates (p_G, α_G) after $N\ t_G$ variables have been assigned by DPLL, see Figure 2. The algorithm then enters the unsat phase and generates 2+p-SAT instances with no solution. Backtracking will appear as soon as a contradiction is detected by DPLL. This may occur at any point along the trajectory (Frieze and Suen, 1996), but no further than the crossing point D with the $\alpha = 1/(1-p)$ line (beyond D, unit-clauses are created at a rate larger than their elimination through unit-propagation, and opposite literals will appear with high probability). Later, massive backtracking enters into play until G is reached again by DPLL. G is indeed the highest backtracking node in the tree, since nodes above G are located in the sat phase and carry 2+p-SAT instances with solutions (Cocco and Monasson, 2001). DPLL will eventually reach a solution S (Figure 1B). This scenario was rigorously established for the whole upper sat phase $\alpha_L < \alpha_0 < \alpha_C$ by Achlioptas and collaborators under the assumption that the largest value of p for which $\alpha_c(p) = 1/(1-p)$ is equal to 2/5 (Achlioptas, Beame and Molloy, 2001c).

2.2. Fast albeit unlikely resolutions.

While in the upper sat phase, search trees almost always look like Figure 1B, they may sometimes consist of a single branch (Figure 1A). We give in the following a lower bound to the probability that the latter case arises, that is, that DPLL finds a solution without ever backtracking. Due to the absence of backtracking, the same probabilistic setting as in past works devoted to GUC may be applied: the search heuristic defines a Markov chain for the evolution of subformulas as more and more variables are set. The major difference is that we are now considering initial densities above α_L, which means that the probabilities of the events we look for are exponentially small (in the number N of variables). In other words, rather than considering the mean resolution trajectory (which unavoidably leads to a contradiction and backtracking), we need to look at large deviations from this trajectory. Notice that, though we focus on a specific algorithm, namely GUC, our approach and the spirit of our results should hold for many others.

Experimental results presented in section 3 provide intuition about runs able to find a solution without backtracking. These are typically runs in which, although the 2-SAT subformula is supercritical ($\alpha > 1/(1-p)$) and a lot ($O(N)$) of unitary clauses are present, no pair of these unitary clauses is contradictory. Such a "miracle" occurs with an exponentially small probability calculated in section 4. The key point of our analysis is that we follow the whole distribution of clause densities in time, and not only their average values (section 2.1). Capturing this additional information requires the introduction of partial differential equations, a more powerful tool than the ordinary differential equations sufficient to study the average behaviour of clause densities.

In principle, there could be even luckier runs than linear ones. However, considering such runs would force us to lose the Markovian nature of the algorithm's random process, a very difficult task at the time being. We shall come back to this point in conclusion.

3. Numerical experiments.

In this section we present some numerical experiments on large instance sizes with a two-fold scope: first, to show the existence of fast resolutions deviating from the high probability scenario exposed above; secondly, to build some intuition on the underlying mechanisms associated to these rare resolutions.

3.1. Instance-to-instance distribution of complexities.

We have first performed some experiments to understand the distribution of instance-to-instance fluctuations of the solving times (Hogg and Williams, 1994; Selman and Kirkpatrick, 1996; Gent and Walsh, 1994). We draw randomly a large number of instances at fixed ratio $\alpha = 3.5$ and size N and, for each of them, run DPLL until a solution is found (a small number of unsat instances can be present and are discarded). We show in Figure 3 the normalized histogram of the logarithms ω of the corresponding complexities $Q = 2^{N\omega}$. The histogram is made of a narrow peak (left side) followed by a wider bump (right side). As N grows, the right peak acquires more and more weight, while the left peak progressively disappears. The abscissa of the center of the right peak gets slightly shifted to the left, but seems to reach a finite value $\omega^* \simeq 0.035$ as $N \to \infty$ (Cocco and Monasson, 2001). This right peaks thus corresponds to the core of exponentially hard resolutions: with high probability resolutions of instances require a time scaling as $2^{N\omega^*}$ as the size of the instance gets larger and larger, in agreement with the discussion of section 2.

On the contrary, the location of the maximum of the left peak seems to vanish as $\log_2(N)/N$ when the size N increases, indicating that the

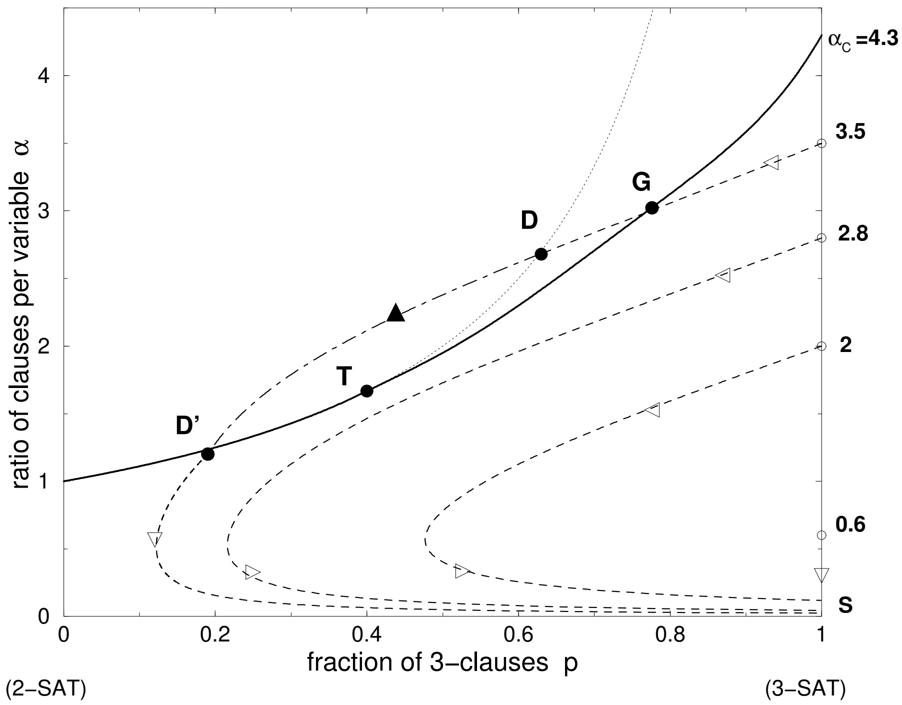

Figure 2. Phase diagram of 2+p-SAT and dynamical trajectories of DPLL for satisfiable instances. The threshold line $\alpha_C(p)$ (bold full line) separates sat (lower part of the plane) from unsat (upper part) phases. Extremities lie on the vertical 2-SAT (left) and 3-SAT (right) axis at coordinates $(p = 0, \alpha_C = 1)$ and $(p = 1, \alpha_C \simeq 4.3)$ respectively. Departure points for DPLL trajectories are located on the 3-SAT vertical axis (empty circles) and the corresponding values of α are explicitly given. Arrows indicate the direction of "motion" along trajectories (dashed curves) parametrized by the fraction t of variables set by DPLL. For small ratios $\alpha < \alpha_L$, branch trajectories remain confined in the sat phase, end in S of coordinates $(1, 0)$, where a solution is found. At α_L ($\simeq 3.003$ for the GUC heuristic, see text), the single branch trajectory hits tangentially the threshold line in T of coordinates $(2/5, 5/3)$. In the range $\alpha_L < \alpha < \alpha_C$, the branch trajectory intersects the threshold line at some point G (that depends on α). With high probability, a contradiction arises before the trajectory crosses the dotted curve $\alpha = 1/(1-p)$ (point D); through extensive backtracking, DPLL later reachs back the highest backtracking node in the search tree (G) and find a solution at the end of a new descending branch, see Figure 1B. With exponentially small probability, the trajectory (dot-dashed curve, full arrow) is able to cross the "dangerous" region where contradictions are likely to occur; it then exits from this contradictory region (point D') and ends up with a solution (lowest dashed curve, light arrow).

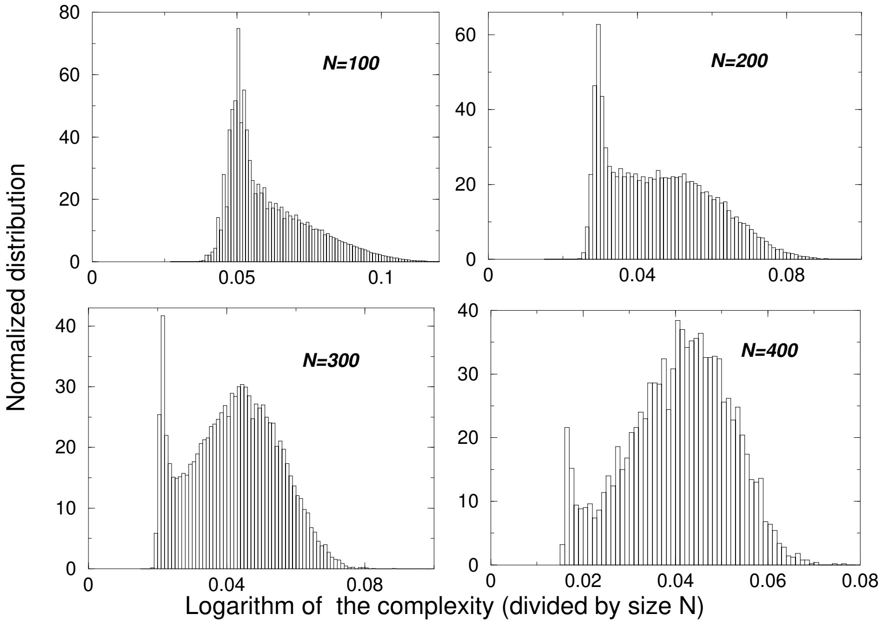

Figure 3. Probability distribution of the logarithm ω of the complexity (base 2, and divided by N) for $\alpha = 3.5$ and for different sizes N. Histograms are normalized to unity and obtained from 400,000 ($N = 100$), 50,000 ($N = 200$), 20,000 ($N = 300$), and 5,000 ($N = 400$) samples

left peak accounts for polynomial (linear) resolutions. We have thus replotted the data shown in Figure 3, changing the scale of the horizontal axis $\omega = \log_2(Q)/N$ into Q/N. Results are shown in Figure 4. We have limited ourself to $Q/N < 1$, the range of interest to analyse the left peak of Figure 3. The maximum of the distribution is located at $Q/N \simeq 0.2 - 0.25$, with weak dependence upon N. The cumulative probability P_{lin} to have a complexity Q less than, or equal to N, *i.e.* the integral of Figure 4 over $0 < Q/N < 1$, decreases very quickly with N. We find an exponential decrease, $P_{lin} = 2^{-N\zeta}$, see Inset of Figure 4[1] The rate $\zeta \simeq 0.011 \pm 0.001$ is determined from the slope of the logarithm of the probability shown in the Inset.

[1] Though one can legitimately wonder if the sizes investigated in experiments allow us to shed light on the asymptotic behaviour of P_{lin}, the linear scaling of $\log P_{lin}$ with $N(\leq 400)$ suggests that this is indeed the case. A theoretical understanding of the finite size effects affecting P_{lin} would be very interesting.

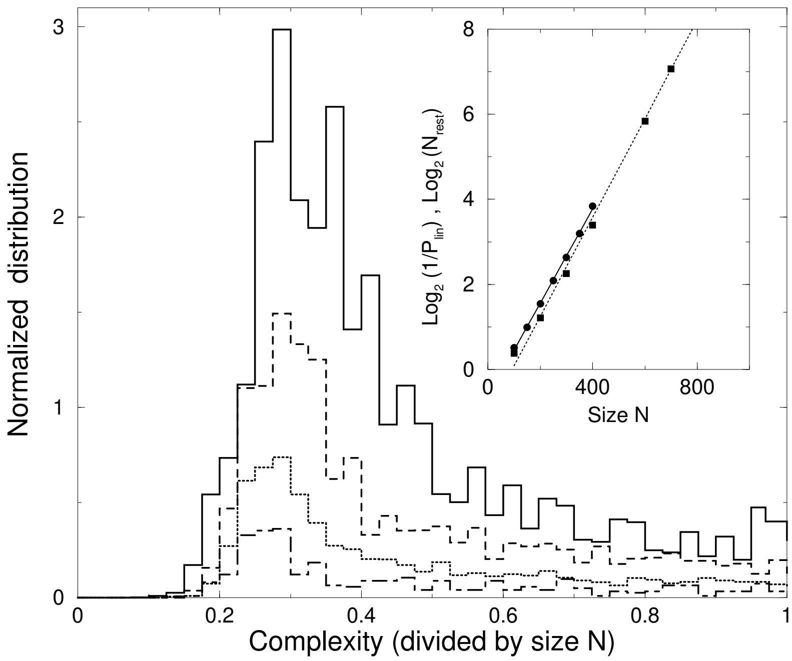

Figure 4. Probability distributions of the complexity Q (divided by the size N) for sizes $N = 100$ (full line), $N = 200$ (dashed line), $N = 300$ (dotted line), $N = 400$ (dashed-dotted line). Distributions are not shown for complexities larger than N. Inset: Minus logarithm of the cumulative probability of complexities smaller or equal to N as a function of N, for sizes ranging from 100 to 400 (full line); logarithm of the number of restarts necessary to find a solution for sizes ranging from 100 to 1000 (dotted line). Slopes are equal to $\zeta = 0.0011$ and $\bar{\zeta} = 0.00115$ respectively.

3.2. Locus of highest backtracking points.

To gain some intuition on the origin of fast, linear resolutions, we have looked for the locus of the highest backtracking nodes G in the search trees. In the infinite size limit, G is located with high probability at the crossing G^* of the resolution trajectory and the critical sat/unsat line (section 2). In Figure 5 we show numerical evidence for the link between complexity and trajectories in the phase diagram for finite instance sizes. We have run 20,000 instances ($\alpha = 3.5, N = 300$), and reported for each of them the coordinates p_G, α_G of the highest backtracking point, and the logarithm ω of the corresponding complexity. Large complexities ($\omega \geq 0.3$, right bump of Figure 3) are associated to points G forming a cloud centered around G^* in the phase diagram of the 2+p-SAT model, while points G related to small complexities

($\omega \leq 0.2$, left peak of Figure 3) are much more scattered in the phase diagram. Notice the strong correlation between the value of ω and the average location of G along the branch trajectory of section 2. In the following we will concentrate on linear resolutions only.

Figure 5 shows that easy resolutions correspond to trajectories capable of trespassing the contradiction line $\alpha = 1/(1-p)$. This, in addition to the linear scaling of the corresponding complexities, indicates that easy resolutions coincide with first descents in the search tree ending with a contradiction located far beyond D in the phase diagram, and then requiring a very limited amount of backtracking before a solution is found. In other words, fast resolutions are ones where the instance generated during resolution is supercritical (the 2-SAT part of the instance has ratio larger than unity) but lucky enough to avoid contradiction.

This statement is supported by the analysis of the number of unit-clause generated during easy resolutions. We have measured the maximal number $(C_1)_{max}$ of unit-clauses generated along the last branch in the tree, leading to the solution S (Figure 1B). We found that $(C_1)_{max}$ scales linearly with N with an extrapolated ratio $(C_1)_{max}/N \simeq 0.022$ for $\alpha = 3.5$. This linear scaling of the number of unit-clauses is an additional proof of the trajectory entering the "dangerous" region $\alpha > 1/(1-p)$ of the phase diagram where unit-clauses accumulate. In presence of a $O(N)$ number of 1-clauses, the probability of survival of the branch (absence of contradictory literals among the unit-clauses) will be exponentially small in N, in agreement with scaling of the left peak weight in Figure 3.

3.3. Run-to-run fluctuations and restart experiments.

We have so far considered the instance-to-instance fluctuations of the complexity, that is the distribution of complexity obtained from one run of DPLL on each of a large number of instances. In Figure 6, we now show the histogram of complexities for a large number of runs on a unique, random instance. After each run, clauses and variables are randomly relabeled to avoid any correlation between different runs. Figure 6 shows that these run-to-run distributions are qualitatively independent of the particular instances, and similar to the instance-to-instance distribution of Figure 6 (Hogg and Williams, 1994; Selman and Kirkpatrick, 1996).

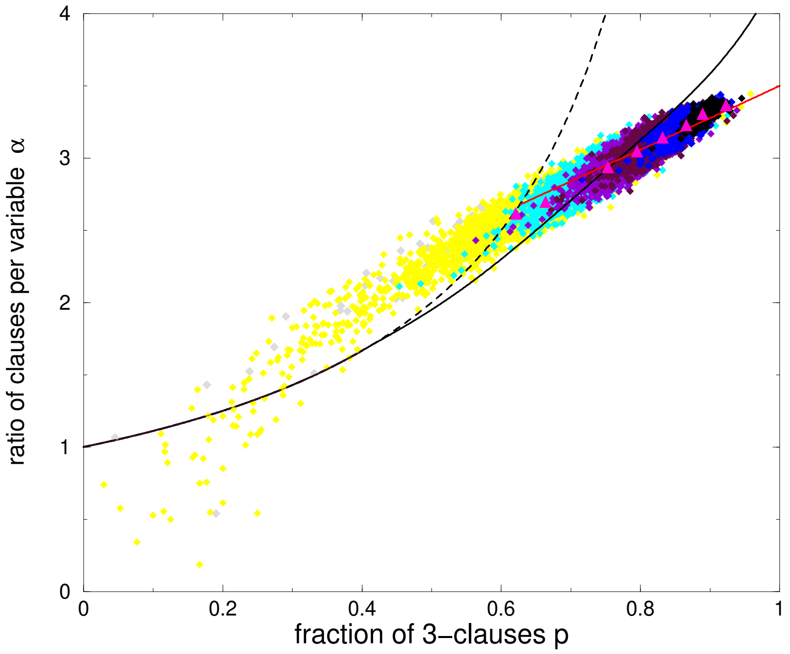

Figure 5. Locus of highest backtracking points G in the phase diagram of the 2+p-SAT model for 20,000 instances with $N = 300$. The bold gray line represent the first branch trajectory for $\alpha = 3.5$. Colors reflect the complexities of the instances, whose logarithms ω range from 0.01 to 0.09, and are divided into 8 intervals of width $\Delta\omega = 0.01$ and increasing darkness. Filled triangles are the center of masses of points G for each of the 8 intervals (the larger ω, the closer to the $p = 1$ axis).

The similarity between run-to-run and instance-to-instance fluctuations for large sizes speaks up for the use of a systematic restart heuristic to speed up resolution: if a solution is not found before N splits, DPLL is stopped and launched again after some random permutations of the variables and clauses. Intuitively, the expected number of restarts necessary to find a solution should indeed be equal to the inverse of the weight of the linear complexity peak in Figure 3, with a resulting total complexity scaling as $N\, 2^{\,0.011\, N}$, and much smaller than the one-run complexity $2^{\,0.035\, N}$ of DPLL (section 2).

We check the above reasoning by measuring the number N_{rest} of restarts performed before a solution is finally reached with the restart heuristic, and averaging $\log_2(N_{res})$ over a large number of random instances. Results are reports in the Inset of Figure 4. The typical number $N_{rest} = 2^{N\bar{\zeta}}$ of required restarts clearly grows exponentially as a function of the size N with a rate $\bar{\zeta} = 0.0115 \pm 0.001$. To the accuracy of the experiments, ζ and $\bar{\zeta}$ coincide as expected.

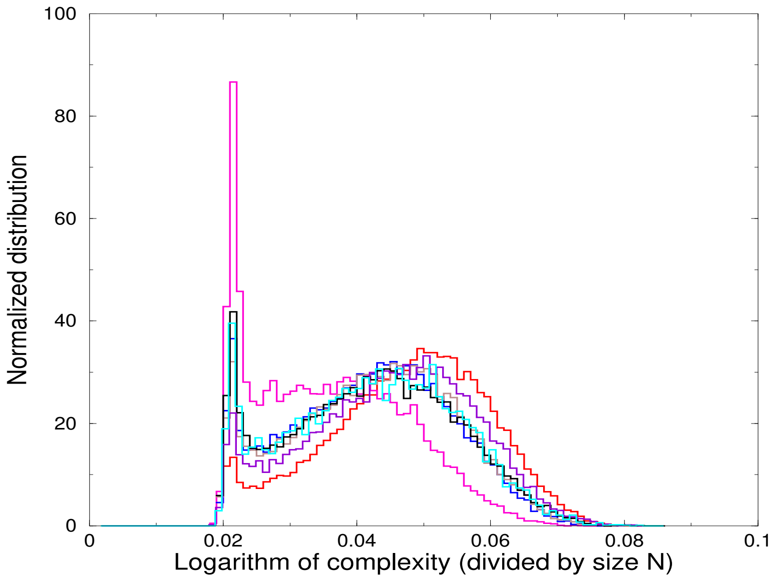

Figure 6. Probability distributions of the logarithm ω of the resolution complexity from 20,000 runs of DPLL. Each one of the five distribution corresponds to one randomly drawn instance of size $N = 300$. The black curve is the instance-to-instance fluctuations of the complexity shown on Figure 3.

4. Large deviation analysis of the first descent in the tree.

4.1. Evolution equation for the instance.

Hereafter we compute the probability $\tilde{P}(C_1, C_2, C_3; T)$ that the first branch of the tree carries an instance with C_j j-clauses ($j = 1, 2, 3$) after T variables have been assigned (and no contradiction has occurred). Let us call $\mathbf{C}$ the vector (C_1, C_2, C_3). P obeys a Markovian evolution

$$\tilde{P}(\mathbf{C}; T+1) = \sum_{\mathbf{C}'} K(\mathbf{C}, \mathbf{C}'; T)\tilde{P}(\mathbf{C}'; T) \quad , \tag{4}$$

where the entries of the transition matrix K are equal to (for the GUC heuristic),

$$K(\mathbf{C}, \mathbf{C}'; T) = \binom{C_3'}{C_3' - C_3} \left(\frac{3}{2M}\right)^{C_3' - C_3} \left(1 - \frac{3}{M}\right)^{C_3} \times$$
$$\sum_{w_2=0}^{C_3'-C_3} \binom{C_3' - C_3}{w_2} \Bigg\{ (1 - \delta_{C_1'}) \sum_{z_2=0}^{C_2'} \binom{C_2'}{z_2} \left(\frac{1}{M}\right)^{z_2} \left(1 - \frac{2}{M}\right)^{C_2' - z_2} \times$$

$$
\begin{aligned}
&\sum_{w_1=0}^{z_2} \binom{z_2}{w_1} \sum_{z_1=0}^{C_1'-1} \binom{C_1'-1}{z_1} \left(\frac{1}{2M}\right)^{z_1} \left(1-\frac{1}{M}\right)^{C_1'-1-z_1} \times \\
&\delta_{C_2-C_2'-w_2+z_2}\, \delta_{C_1-C_1'-w_1+z_1+1} + \delta_{C_1'} \sum_{z_2=0}^{C_2'-1} \binom{C_2'-1}{z_2} \left(\frac{1}{M}\right)^{z_2} \times \\
&\left.\left(1-\frac{2}{M}\right)^{C_2'-1-z_2} \sum_{w_1=0}^{z_2} \binom{z_2}{w_1}\, \delta_{C_2-C_2'-w_2+z_2+1}\, \delta_{C_1-w_1} \right\} , \qquad (5)
\end{aligned}
$$

and δ_C denotes the Kronecker delta function: $\delta_C = 1$ if $C=0$, $\delta_C = 0$ otherwise. $M \equiv N-T$ is the number of unassigned variables. $K(\mathbf{C},\mathbf{C}';T)$ is the probability that an instance with clause vector $\mathbf{C}'$ is turned into an instance with clause vector $\mathbf{C}$ when the $(T+1)^{th}$ variable is assigned.

We then define the generating function $P(\mathbf{y}\,;T\,)$ of the probabilities $\tilde{P}(\,\mathbf{C}\,;T\,)$ where $\mathbf{y} \equiv (y_1, y_2, y_3)$, through

$$
P(\,\mathbf{y}\,;T\,) = \sum_{\mathbf{C}} e^{\,\mathbf{y}\cdot\mathbf{C}}\, \tilde{P}(\,\mathbf{C}\,,T\,) \qquad (6)
$$

where $\cdot$ denotes the scalar product. Evolution equation (4) can be rewritten in term of the generating function P,

$$
\begin{aligned}
P(\,\mathbf{y}\,;T+1\,) \;=\; & e^{-g_1(\mathbf{y})}\, P(\,\mathbf{g}(\mathbf{y})\,;T\,) + \\
& \left(e^{-g_2(\mathbf{y})} - e^{-g_1(\mathbf{y})}\right) P(\,-\infty,\, g_2(\mathbf{y}),\, g_3(\mathbf{y})\,;T\,) \qquad (7)
\end{aligned}
$$

where $\mathbf{g}$ is a vectorial function of argument $\mathbf{y}$ whose components read

$$
\begin{aligned}
g_1(\mathbf{y}) &= y_1 + \ln\left[1 + \frac{1}{N-T}\left(\frac{e^{-y_1}}{2} - 1\right)\right] \\
g_2(\mathbf{y}) &= y_2 + \ln\left[1 + \frac{2}{N-T}\left(\frac{e^{-y_2}}{2}\,(1+e^{y_1}) - 1\right)\right] \\
g_3(\mathbf{y}) &= y_3 + \ln\left[1 + \frac{3}{N-T}\left(\frac{e^{-y_3}}{2}\,(1+e^{y_2}) - 1\right)\right] \qquad (8)
\end{aligned}
$$

We now solve equation (7) by making some hypothesis on the scaling behaviour of P for large sizes.

4.2. Hypothesis for the large N scaling of the probability.

Calculations leading to equation (7) are rigorous and do not rely on any independence assumption e.g. the one made in our study of the average-time complexity of DPLL in presence of backtracking (Cocco and Monasson, 2001). We shall now make some hypothesis on $\tilde{P}, P$ that we believe to be correct in the large size N limit, but without providing rigorous proofs for their validity. Our approach, common in statistical mechanics, may be seen as a practical way to establish conjectures.

Use of differential equation techniques require to deal with densities of clauses, and thus to make several assumption on the order of magnitude of the clauses populations. First, each time DPLL assigns variables through splitting or unit-propagation, the numbers C_j of clauses of length j undergo $O(1)$ changes. It is thus sensible to assume that after a number $T = t\,N$ of variables have been assigned, the densities $c_j = C_j/N$ of clauses have been modified by $O(1)$. This translates into a scaling Ansatz for the probability $\tilde{P}$,

$$\tilde{P}(\mathbf{C};T) = e^{N\,\tilde{\varphi}(c_1,c_2,c_3;t)} \qquad (\tilde{\varphi} \leq 0) \tag{9}$$

up to non exponential in N corrections. From equations (6) and (9), we obtain the following scaling hypothesis for the generating function P,

$$P(\mathbf{y};T) = e^{N\,\varphi(\,\mathbf{y}\,;t\,)} \tag{10}$$

up to non exponential in N terms. Notice that φ and $\tilde{\varphi}$ are simply related to each other through Legendre transform,

$$\varphi(\,\mathbf{y}\,;t\,) \;=\; \max_{\mathbf{c}} \left[\tilde{\varphi}(\,\mathbf{c}\,;t\,) + \mathbf{y}\cdot\mathbf{c}\right] \quad , \tag{11}$$

$$\tilde{\varphi}(\,\mathbf{c}\,;t\,) \;=\; \min_{\mathbf{y}} \left[\varphi(\,\mathbf{y}\,;t\,) - \mathbf{y}\cdot\mathbf{c}\right] \quad . \tag{12}$$

In particular, $\varphi(\mathbf{y} = \mathbf{0};t)$ is the logarithm of the probability (divided by N) that the first branch has not been hit by any contradiction after a fraction t of variables have been assigned. The most probable values of the densities $c_j(t)$ of j-clauses are then obtained from the partial derivatives of $\varphi(\,\mathbf{y}\,;t\,)$ in $\mathbf{y} = \mathbf{0}$: $c_j(t) = \partial\varphi/\partial y_j(\mathbf{y} = \mathbf{0})$.

We now present the partial differential equations (PDE) obeyed by φ. Two cases must be distinguished: the number C_1 of unit-clauses may be bounded ($C_1 = O(1), c_1 = o(1)$), or of the order of the instance size ($C_1 = \Theta(N), c_1 = \Theta(1)$).

4.3. Case $C_1 = O(1)$: a large deviation analysis around Frieze and Suen's result.

When DPLL starts running on a 3-SAT instance, very few unit-clauses are generated and splittings occur frequently. In other words, the probability that $C_1 = 0$ is strictly positive when N gets large. Consequently, both terms on the right hand side of (7) are of the same order, and we make the hypothesis that φ does not depend on y_1: $\varphi(y_1,y_2,y_3;t) = \varphi(y_2,y_3;t)$. This hypothesis simply expresses that $c_1 = \partial\varphi/\partial y_1$ identically vanishes.

Inserting expression (10) into the evolution equation (7), we find[2]

$$\begin{aligned}\frac{\partial\varphi}{\partial t}(y_2,y_3;t) = -y_2 \ + \ 2\,\gamma(y_2,y_2;t)\,\frac{\partial\varphi}{\partial y_2}(y_2,y_3;t) \\ + \ 3\,\gamma(y_2,y_3;t)\,\frac{\partial\varphi}{\partial y_3}(y_2,y_3;t)\end{aligned} \tag{13}$$

where function γ is defined through,

$$\gamma(u,v;t) = \frac{1}{1-t}\left(\frac{e^{-v}}{2}\,(1+e^{u}) - 1\right) \quad . \tag{14}$$

PDE (13) together with initial condition $\varphi(\mathbf{y};t=0) = \alpha_0\, y_3$ (where α_0 is the ratio of clauses per variable of the 3-SAT instance) can be solved exactly with the resulting expression,

$$\begin{aligned}\varphi(y_2,y_3;t) \ &= \ \alpha_0 \ln\left[1 + (1-t)^3\left(e^{y_3} - \frac{3}{4}e^{y_2} - \frac{1}{4}\right) + \frac{3(1-t)}{4}(e^{y_2}-1)\right] \\ &+ \ (1-t)\,y_2\,e^{y_2} + (1-t)(e^{y_2}-1)\ln(1-t) \\ &- \ (e^{y_2} + t - t\,e^{y_2})\ln(e^{y_2} + t - t\,e^{y_2})\end{aligned} \tag{15}$$

Chao and Franco, and Frieze and Suen's analysis of the GUC heuristic may be recovered when $y_2 = y_3 = 0$ as expected. It is an easy check that $\varphi(y_2 = 0, y_3 = 0; t) = 0$ *i.e.* the probability of survival of the branch is not exponentially small in N (Frieze and Suen, 1996), and that the derivatives $c_2(t), c_3(t)$ of $\varphi(y_2,y_3;t)$ with respect to y_2 and y_3 coincide with the solutions of (1).

In addition, our calculation provides also a complete description of rare deviations of the resolution trajectory from its highly probable locus shown on Figure 2. As a simple numerical example, consider DPLL acting on a 3-SAT instance of ratio $\alpha_0 = 3.5$. Once e.g. $t = 20\%$ of variables have been assigned, the densities of 2- and 3-clauses are with high probability equal to $c_2 \simeq 0.577$ and $c_3 \simeq 1.792$ respectively. Expression (15) gives access to the exponentially small probabilities that c_2 and c_3 differ from their most probable values. For instance, choosing $y_2 = -0.1, y_3 = 0.05$, we find from (15) and (12) that there

[2] PDE (13) is correct in the major part of the y_1, y_2, y_3 space and, in particular, in the vicinity of $\mathbf{y} = \mathbf{0}$ we focus on in this paper. It has however to be to modified in a small region of the y_1, y_2, y_3 space; a complete analysis of this case is not reported here but may be easily reconstructed along the lines of Appendix A in (Cocco and Monasson, 2001b).

is a probability $e^{-0.00567N}$ that $c_2 = 0.504$ and $c_3 = 1.873$ for the same fraction $t = 0.2$ of eliminated variables. By scanning all the values of y_2, y_3 we can obtain a complete description of large deviations from Frieze and Suen's result[3].

The assumption $C_1 = O(1)$ breaks down for the most probable trajectory at some fraction t_D e.g. $t_D \simeq 0.308$ for $\alpha_0 = 3.5$ at which the trajectory hits point D on Figure 2. Beyond D, 1-clauses accumulate and the probability of survival of the first branch is exponentially small in N.

4.4. Case $C_1 = O(N)$: passing through the "dangerous" region.

When the number of unit-clauses becomes of the order of N, variables are almost surely assigned through unit-propagation. The first term on the right hand side of equation (7) is now exponentially dominant with respect to the second one. The density of 1-clauses is strictly positive, and φ depends on y_1. We then obtain the following PDE,

$$\begin{aligned}
\frac{\partial \varphi}{\partial t}(y_1, y_2, y_3; t) &= -y_1 + \gamma(-\infty, y_1; t)\, \frac{\partial \varphi}{\partial y_1}(y_1, y_2, y_3; t) \\
&+ 2\,\gamma(y_1, y_2; t)\, \frac{\partial \varphi}{\partial y_2}(y_1, y_2, y_3; t) \\
&+ 3\,\gamma(y_2, y_3; t)\, \frac{\partial \varphi}{\partial y_3}(y_1, y_2, y_3; t)
\end{aligned} \tag{16}$$

with $\gamma(u, v; t)$ given by equation (14). When $y_1 = y_2 = y_3 = 0$, equation (16) simplifies to

$$\frac{dz}{dt}(t) = -\frac{c_1(t)}{2(1-t)} \qquad , \tag{17}$$

where $c_1(t)$ is the most probable value of the density of unit-clauses, and $z(t)$ is the logarithm of the probability that the branch has not encountered any contradiction (divided by N). The interpretation of (17) is transparent. Each time a literal is assigned through unit-propagation, there is a probability $(1 - 1/2(N-T))^{C_1 - 1} \simeq e^{-c_1/2/(1-t)}$ that no contradiction occurs. The right hand side of (17) thus corresponds to the rate of decay of z with "time" t.

We have not been able to solve analytically PDE (16), and have resorted to an expansion of φ in powers of $\mathbf{y}$. To k^{th} order, we approximate

[3] Though we are not concerned here with sub-exponential (in N) corrections to probabilities, let us mention that it is also possible to calculate the probability of split ($C_1 = 0$) per unit of time, extending Frieze and Suen's result (1996) to $\mathbf{y} \neq \mathbf{0}$.

Table I. Results at different orders k of approximation for $\alpha_0 = 3.5$: logarithm ζ of the probability that the first branch is not hit by any contradiction, maximal density $(c_1)_{max}$ of unit-clauses ever reached, fraction of eliminated variables $t_{D'}$ and coordinates $p_{D'}, \alpha_{D'}$ at point D' *i.e.* when the number of unit-clauses ceases to be $O(N)$, complexity ratio $\gamma = Q/N$ of the corresponding linear resolution.

order	ζ	$(c_1)_{max}$	$t_{D'}$	$p_{D'}$	$\alpha_{D'}$	γ
1	.0384	.0502	.8878	.0804	.5477	.1720
2	.0036	.0121	.6553	.2707	1.575	.1990
3	.0098	.0227	.7495	.1901	1.201	.2069
4	.0098	.0226	.7483	.1911	1.206	.2069

the solution of (16) by a polynomial of total degree k,

$$\varphi^{(k)}(\mathbf{y};t) = \sum_{e_1+e_2+e_3\leq k} \varphi^{(k)}_{e_1,e_2,e_3}(t)\; y_1^{e_1}\; y_2^{e_2}\; y_3^{e_3} \tag{18}$$

and insert (18) on the right hand side of (16)[4]. We collect on the l.h.s. the terms of degrees $\leq k$ and obtain a set of $\mathcal{N}_k = (k+3)(k+2)(k+1)/6$ first order coupled linear ODEs for the coefficients $\varphi^{(k)}_{e_1,e_2,e_3}(t)$ of the polynomial (18). This approximation gets better and better as k increases at a cost of more and more coupled ODEs to be solved. The initial conditions for these ODEs are chosen to match the expansion of the exact solution (15) at time t_D.

At the lowest order ($k=1$), we find a set of four coupled equations for $z^{(1)}(t) \equiv \varphi^{(1)}_{0,0,0}(t), c_1^{(1)}(t) \equiv \varphi^{(1)}_{1,0,0}(t), c_2^{(1)}(t) \equiv \varphi^{(1)}_{0,1,0}(t), c_3^{(1)}(t) \equiv \varphi^{(1)}_{0,0,1}(t)$ that read

$$\begin{aligned}
\frac{dc_1^{(1)}(t)}{dt} &= -1 + \frac{c_2^{(1)}(t)}{1-t} \\
\frac{dc_2^{(1)}(t)}{dt} &= -\frac{2\,c_2^{(1)}(t)}{1-t} + \frac{3\,c_3^{(1)}(t)}{2(1-t)} \\
\frac{dc_3^{(1)}(t)}{dt} &= -\frac{3\,c_3^{(1)}(t)}{1-t}
\end{aligned}$$

[4] This approximation is unfortunately not as innocuous as it may appear at first sight. It amounts to consider that φ may be expanded in series around $\mathbf{y} = \mathbf{0}$, an assumption that would turn out to be false if non analytic contributions e.g. $|y_1|$ were present.

$$\frac{dz^{(1)}(t)}{dt} = -\frac{c_1^{(1)}(t)}{2(1-t)} \quad , \tag{19}$$

together with the initial conditions $c_1^{(1)}(t_D) = z^{(1)}(t_D) = 0, c_2^{(1)}(t_D) = 1-t_D, c_2^{(1)}(t_D) = \alpha_0(1-t_D)^3$, with t_D uniquely determined from α_0. The solution of (19) for $\alpha_0 = 3.5$ shows that c_1 first increases and reaches its top value $(c_1^{(1)})_{max} \simeq 0.05$. It then decreases and vanishes at $t_{D'}^{(1)} \simeq 0.89$, where the trajectory exits the "dangerous" region where contradictions occurs with high probability (Figure 2). The probability of this event scales as $2^{-N\zeta^{(1)}}$ for large N, with $\zeta^{(1)} = -z^{(1)}(t_{D'}^{(1)})/\ln 2 \simeq 0.038$. The end of the resolution trajectory obeys Chao and Franco's equations (1).

Results improve when going to higher orders in k, see Table I. No sensible difference can be found between $k=3$ and $k=4$ results, which are shown in Figure 7. The calculated values of $\zeta \simeq 0.01, (c_1)_{max} \simeq 0.022$ and $\gamma \simeq 0.21$ are in very good agreement with the numerical experiments of section 3.

We report on Figure 8 the experimental and theoretical values of ζ found over the whole range $\alpha_L \leq \alpha_0 \leq \alpha_C$. Note the very good agreement between our quantitative theory and simulations, which supports the scaling hypothesis made above.

5. Conclusions.

In this work, we have presented experimental and analytical evidence that DPLL algorithms should aim at being lucky, not thorough. To do so, we have studied deviations from the typical (occurring with high probability) resolution complexity of satisfiable random 3-SAT instances using DPLL algorithm with a simple splitting heuristic (GUC) (Chao and Franco, 1986; Chao and Franco, 1990). For ratios α of clauses per variable in the range $\alpha_L = 3.003 < \alpha < \alpha_C$, complexity grows almost surely exponentially with the size N of the instance, but resolution may very rarely (*i.e.* with an exponentially small probability) require a polynomial (linear) computational effort only. These linear resolutions correspond to search tree reducing to a single branch essentially (Figure 1A), and can be visualized as trajectories that cross the unsat phase of the Figure 2 diagram without being stopped by any contradiction. Our approach allowed us to calculate the large deviations from typical resolutions, and the exponent ζ of the probability $\tilde{P} \sim 2^{-\zeta N}$ of linear resolutions. Our theoretical calculation predicts for instance that the exponent corresponding to random 3-SAT instances with ratio $\alpha = 3.5$ equals $\zeta \simeq 0.01$, in very good agreement with the values extrapolated

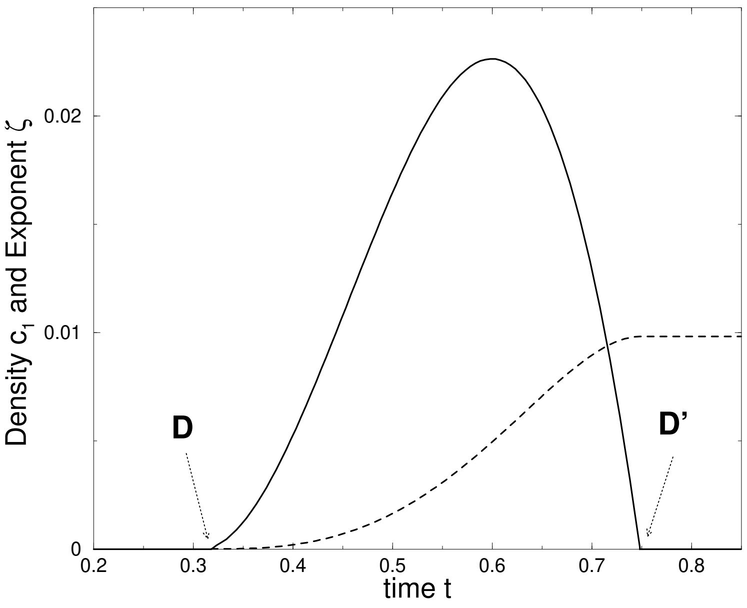

Figure 7. Density c_1 of unitary clauses (full line) and logarithm z (17) of the probability of the absence of contradiction along the first branch (dashed line) as a function of time t (fraction of assigned variables) for an initial ratio $\alpha = 3.5$. The density of unit clauses is positive between points D and D' along the branch trajectory of Figure 2; z is null before the trajectory reaches D, and constant and equal to the exponent ζ beyond D'. These curves are the output of the $k = 4$ approximate resolution of PDE (16), see text.

from the histogram of resolution time on different instances (instance-to-instance distribution of complexities) and the value $\bar{\zeta}$ extrapolated from the number of restarts necessary to solve one random instance (Inset of Figure 4).

The computational effort to find a solution with the systematic restart procedure, $N_{rest} \sim 1/P_{lin} \sim 2^{N\zeta}$ turns out to be exponentially smaller than the typical time to find a solution $2^{N\omega}$ without restart (e.g. $\omega = 0.035$ for $\alpha = 3.5$). Our calculation gives thus some theoretical support to the use of restart-like procedures (see also (Montanari and Zecchina, 2002) for recent theoretical results), empirically known to speed up considerably resolutions (Dubois et al., 1993; Gomes et al., 2000). To be more concrete, while, without restarts, we were able to solve with DPLL algorithm instances with 500 variables in about one day of CPU (for $\alpha = 3.5$), the restart procedure allowed us to solve

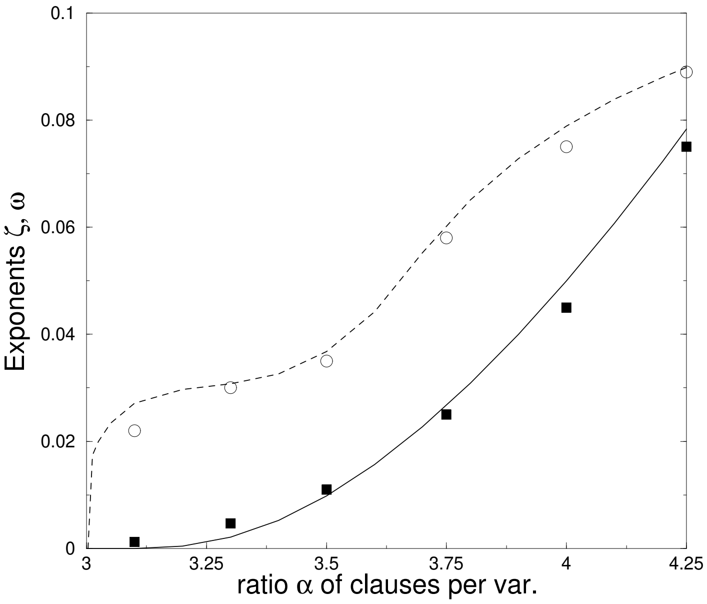

Figure 8. Exponent ζ of the linear resolution probability (simulations: filled squares, theory: full line), and exponent ω of the typical complexity (simulations: empty circles, theory from (Cocco, Monasson 2001b): dotted line), as a function of the clause per variable ratio α.

instances with 1000 variables in 15 minutes with the same computer and splitting heuristic (GUC).

The present work suggests that the cut-off time, at which the search is halted and restarted, need not be precisely tuned but is simply given by the size of the instance. This conclusion could be generic and apply to other combinatorial decision problems and other heuristics. More precisely, if a combinatorial problem admits some efficient (polynomial) search heuristic for some values of control parameter (e.g. the ratio α here, or the average adjacency degree for the coloring problem of random graphs), there might be an exponentially small probability that the heuristic is still successful (in polynomial time) in the range of parameters where resolution almost surely requires massive backtracking and exponential effort. When the decay rate of the polynomial time resolution probability ζ is smaller than the growth rate ω of the typical exponential resolution time, restart procedures with a cut-off in the search equal to a polynomial of the instance size will lead to an exponential speed up of resolutions.

In principle, one could not rule out the existence of even luckier runs than linear ones. For instance, there could exist exponentially long (complexity $2^{\omega' N}$ with $0 < \omega' < \omega$) and rare (probability $2^{-\zeta' N}$ with $0 < \zeta' < \zeta$) runs with $\omega' + \zeta' < \zeta$. If so, $2^{\omega' N}$ would be a better cut-off for restart than N. A recent analysis of the distribution of exponentially long resolutions indicates this is not so for the problem of the vertex covering of random graphs, and that the optimal cut-off for restarts is indeed the instance size itself (Montanari and Zecchina, 2002).

It would be interesting to extend the previous approach to more sophisticated and powerful search e.g. satz of chaff heuristics. It is however not clear how a full analytical study could be worked out without resorting to approximate expressions for the transition matrix. Another natural extension of the present work would be to focus on other decisions problems e.g. graph coloring for which the high probability behaviour of simple heuristics is well understood (Achlioptas and Molloy, 1997).

Acknowledgements

We thank D. Achlioptas and the referees for useful suggestions to improve the presentation of this work. R.M. is supported in part by the ACI Jeunes Chercheurs “Algorithmes d’optimisation et systèmes désordonnés quantiques” from the French Ministry of Research.

References

Achlioptas, D. and Molloy, M. Analysis of a List-Coloring Algorithm on a Random Graph, in *Proceedings of FOCS 97*, p. 204-212 (1997).

Achlioptas, D., Kirousis, L., Kranakis, E. and Krizanc, D. Rigorous results for random (2+p)-SAT, *Theoretical Computer Science* **265**, 109-129 (2001).

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

Achlioptas, D., Beame, P. and Molloy, M. A Sharp Threshold in Proof Complexity. in *Proceedings of STOC 01*, p.337-346 (2001).

Beame, P., Karp, R., Pitassi, T. and Saks, M. *ACM Symp. on Theory of Computing (STOC98)*, 561–571 Assoc. Comput. Mach., New York (1998).

Chao, M.T. and Franco, J. Probabilistic analysis of two heuristics for the 3-satisfiability problem, *SIAM Journal on Computing* **15**, 1106-1118 (1986).

Chao, M.T. and Franco, J. Probabilistic analysis of a generalization of the unit-clause literal selection heuristics for the k-satisfiability problem, *Information Science* **51**, 289–314 (1990).

Chvàtal, V. and Szmeredi, E. Many hard examples for resolution, *Journal of the ACM* **35**, 759–768 (1988).

Coarfa, C., Dernopoulos, D.D., San Miguel Aguirre, A., Subramanian, D. and Vardi, M.Y. Random 3-SAT: The plot thickens. In R. Dechter, editor, *Proc. Principles and Practice of Constraint Programming (CP'2000)*, Lecture Notes in Computer Science 1894, 143-159 (2000);

Cocco, S. and Monasson R. Trajectories in phase diagrams, growth processes and computational complexity: how search algorithms solve the 3-Satisfiability problem, *Phys. Rev. Lett.* **86**, 1654 (2001)

Cocco, S. and Monasson, R. Analysis of the computational complexity of solving random satisfiability problems using branch and bound search algorithms, *Eur. Phys. J. B* **22**, 505 (2001).

Cocco, S. and Monasson R. Exponentially hard problems are sometimes polynomial, a large deviation analysis of search algorithms for the random satisifiability problem, and its application to stop-and-restart resolutions, *Phys. Rev. E* **66**, 037101 (2002).

Crawford, J. and Auton, L. Experimental Results on the Cross-Over Point in Satisfiability Problems, *Proc. 11th Natl. Conference on Artificial Intelligence (AAAI-93),* 21–27, The AAAI Press / MIT Press, Cambridge, MA (1993); *Artificial Intelligence* **81** (1996).

Davis, M., Logemann, G. and Loveland, D. A machine program for theorem proving. *Communications of the ACM* **5**, 394-397 (1962).

Dubois, O., Andre, P., Boufkhad, Y. and Carlier, J. SAT versus UNSAT, *DIMACS Series in Discrete Math. and Computer Science*, 415–436 (1993).

Dubois, O., Boufkhad, Y. and Mandler, J. Typical random 3-SAT formulae and the satisfiability threshold. *SODA*, p. 126-127 (2000).

Dubois, O., Monasson, R., Selman, B. and Zecchina, R. (eds) Phase transitions in combinatorial problems. *Theor. Comp. Sci.* **265** (2001).

Franco, J. Results related to thresholds phenomena research in satisfiability: lower bounds. *Theor. Comp. Sci.* **265**, 147–157 (2001).

Friedgut, E. Sharp thresholds of graph properties, and the k-sat problem, *Journal of the A.M.S.* **12**, 1017 (1999).

Frieze, A. and Suen, S. Analysis of two simple heuristics on a random instance of k-SAT, *Journal of Algorithms* **20**, 312–335 (1996).

Gent, I.P. and Walsh, T. Easy problems are sometimes hard, *Artificial Intelligence* **70**, 335-345 (1994).

Gent, I., van Maaren, H. and Walsh, T. (eds), SAT2000: Highlights of Satisfiability Research in the Year 2000, *Frontiers in Artificial Intelligence and Applications*, vol. 63, IOS Press, Amsterdam (2000).

Gomes, C.P., Selman, B., Crato, N. and Kautz, H. *J. Automated Reasoning* **24**, 67 (2000).

Gu, J., Purdom, P.W., Franco, J. and Wah, B.W. Algorithms for satisfiability (SAT) problem: a survey. *DIMACS Series on Discrete Mathematics and Theoretical Computer Science* **35**, 19-151, American Mathematical Society (1997).

Hartmann, A. and Weigt, M. Typical solution time for a vertex-covering algorithm on finite-connectivity random graphs, *Phys. Rev. Lett.* **86**, 1658 (2001).

Hogg, T. and Williams, C.P. The hardest constraint problems: a double phase transition, *Artificial Intelligence* **69**, 359-377 (1994).

Hogg, T., Huberman, B.A. and Williams, C. (eds) Frontiers in problem solving: phase transitions and complexity. *Artificial Intelligence* **81** I & II (1996).

Kaporis, A.C., Kirousis, L.M. and Lalas, E.G. The Probabilistic Analysis of a Greedy Satisfiability Algorithm. *ESA*, p. 574-585 (2002).

Kirkpatrick, S., and Selman, B. Critical Behavior in the Satisfiability of Random Boolean Expressions. *Science* **264**, 1297–1301 (1994).

Mitchell, D., Selman, B. and Levesque, H. Hard and Easy Distributions of SAT Problems, *Proc. of the Tenth Natl. Conf. on Artificial Intelligence (AAAI-92)*, 440-446, The AAAI Press / MIT Press, Cambridge, MA (1992).

Monasson, R., Zecchina, R., Kirkpatrick, S., Selman, B. and Troyansky, L. Determining computational complexity from characteristic 'phase transitions'. *Nature* **400**, 133–137 (1999); 2+p-SAT: Relation of Typical-Case Complexity to the Nature of the Phase Transition, *Random Structure and Algorithms* **15**, 414 (1999).

Montanari, A., and Zecchina, R. Boosting search by rare events, *Phys. Rev. Lett.* **88**, 178701 (2002).

Selman, B. and Kirkpatrick, S. Critical behavior in the computational cost of satisfiability testing, *Artificial Intelligence* **81**, 273-295 (1996).