

Optimization hardness as transient chaos in an analog approach to constraint satisfaction

Mária Ercsey-Ravasz^{1,2*} and Zoltán Toroczkai^{1,2,3*}

Boolean satisfiability¹ (k -SAT) is one of the most studied optimization problems, as an efficient (that is, polynomial-time) solution to k -SAT (for $k \geq 3$) implies efficient solutions to a large number of hard optimization problems^{2,3}. Here we propose a mapping of k -SAT into a deterministic continuous-time dynamical system with a unique correspondence between its attractors and the k -SAT solution clusters. We show that beyond a constraint density threshold, the analog trajectories become transiently chaotic⁴⁻⁷, and the boundaries between the basins of attraction⁸ of the solution clusters become fractal⁷⁻⁹, signalling the appearance of optimization hardness¹⁰. Analytical arguments and simulations indicate that the system always finds solutions for satisfiable formulae even in the frozen regimes of random 3-SAT (ref. 11) and of locked occupation problems¹² (considered among the hardest algorithmic benchmarks), a property partly due to the system's hyperbolic^{4,13} character. The system finds solutions in polynomial continuous time, however, at the expense of exponential fluctuations in its energy function.

Boolean satisfiability¹ (k -SAT, $k \geq 3$) is the quintessential constraint-satisfaction problem, lying at the basis of many decision, scheduling, error-correction and computational applications. k -SAT is in NP (refs 1-3), that is its solutions are efficiently (polynomial time) checkable, but no efficient (polynomial time) algorithms are known to compute those solutions. If such algorithms would be found for k -SAT, all NP problems would be efficiently computable, because k -SAT is NP-complete^{2,3}.

In k -SAT there are given N Boolean variables $\{x_1, \dots, x_N\}$, $x_i \in \{0, 1\}$ and M clauses (constraints), each clause being the disjunction (OR, denoted as \vee) of k variables or their negation (\bar{x}). One has to find an assignment of the variables such that all clauses (called collectively as a formula) are satisfied (TRUE = '1'). When the number of constraints is small, it is easy to find solutions, whereas for too many constraints it is easy to decide that the formula is unsatisfiable (UNSAT). Deciding satisfiability, in the 'intermediate range', however, can be very hard: the worst-case complexity of all known algorithms for k -SAT is exponential in N .

Inspired by the mechanisms of information processing in biological systems, analog computing received increasing interest from both theoretical¹⁴⁻¹⁶ and engineering communities¹⁷⁻²¹. Although the theoretical possibility of efficient computation using chaotic dynamical systems has been shown previously¹⁵, nonlinear dynamical systems theory has not been exploited for NP-complete problems in spite of the fact that, as shown previously¹⁹⁻²¹, k -SAT can be formulated as a continuous global optimization problem¹⁹, and even cast as an analog dynamical system^{20,21}.

Here we present a continuous-time dynamical system for k -SAT, with a dynamics that is rather different from previous approaches.

Let us introduce the continuous variables¹⁹ $s_i \in [-1, 1]$, such that $s_i = -1$ if the i th variable (x_i) is FALSE and $s_i = 1$ if it is TRUE. We define $c_{mi} = 1$ for the direct form (x_i), $c_{mi} = -1$ for the negated form (\bar{x}_i) and $c_{mi} = 0$ for the absence of the i th variable from clause m . Defining the constraint function $K_m(\mathbf{s}) \equiv 2^{-k} \prod_{i=1}^N (1 - c_{mi}s_i)$ corresponding to clause m , we have $K_m \in [0, 1]$ and $K_m = 0$ if and only if clause m is satisfied. The goal would be to find a solution \mathbf{s}^* with $s_i^* \in \{-1, 1\}$ to $E(\mathbf{s}^*) = 0$, where E is the energy function $E(\mathbf{s}) = \sum_{m=1}^M K_m(\mathbf{s})^2$. If such \mathbf{s}^* exists, it will be a global minimum for E and a solution to the k -SAT problem. However, finding \mathbf{s}^* by a direct minimization of $E(\mathbf{s})$ will typically fail owing to non-solution attractors trapping the search dynamics. To avoid such traps, here we define a modified energy function $V(\mathbf{s}, \mathbf{a}) = \sum_{m=1}^M a_m K_m(\mathbf{s})^2$, using auxiliary variables $a_m \in (0, \infty)$ similar to Lagrange multipliers^{20,21}. Let us denote by \mathcal{H}_N the continuous domain $[-1, 1]^N$. Its boundary is the N -hypercube $Q_N = \partial\mathcal{H}_N$ with vertex set $\mathcal{V}_N = \{-1, 1\}^N \subset Q_N$. The set of solutions for a given k -SAT formula, called solution space, occupies a subset of \mathcal{V}_N . Solution clusters are formed by solutions that can be connected through single-variable flips, always staying within satisfying assignments²². Clearly, $V \geq 0$ in $\Omega \equiv \mathcal{H}_N \times (0, \infty)^M$, and $V(\mathbf{s}, \mathbf{a}) = 0$ within \mathcal{V}_N if and only if $\mathbf{s} = \mathbf{s}^* \in \mathcal{V}_N$ is a k -SAT solution, for any $\mathbf{a} \in (0, \infty)^M$. We now introduce a continuous-time dynamical system on Ω through:

$$\frac{ds_i}{dt} = (-\nabla_s V(\mathbf{s}, \mathbf{a}))_i = \sum_{m=1}^M 2a_m c_{mi} K_{mi}(\mathbf{s}) K_m(\mathbf{s}), \quad i = 1, \dots, N \quad (1a)$$

$$\frac{da_m}{dt} = a_m K_m(\mathbf{s}), \quad m = 1, \dots, M \quad (1b)$$

where ∇_s is the gradient operator with respect to \mathbf{s} and $K_{mi} = K_m/(1 - c_{mi}s_i)$. The initial conditions for \mathbf{s} are arbitrary $\mathbf{s}(0) \in \mathcal{H}_N$; however, for \mathbf{a} they have to be strictly positive, $a_m(0) > 0$ (for example, $a_m(0) = 1$). The k -SAT solutions $\mathbf{s}^* \in \mathcal{V}_N$ are fixed points of (1), for any $\mathbf{a} \in (0, \infty)^M$. The k -SAT solution clusters are spanning piecewise compact, connected sets in Q_N , and every point in them is a fixed point of (1) (Supplementary Section SA). System (1) has a number of key properties (see Supplementary Information). First, the dynamics in \mathbf{s} stays confined to \mathcal{H}_N . Second, the k -SAT solutions $\mathbf{s}^* \in \mathcal{V}_N$ are attractive fixed points of (1). In particular, every point \mathbf{s} from the orthant of a k -SAT solution \mathbf{s}^* with the property $|\mathbf{s}|^2 \geq N - 1 + (k-1)^2/(k+1)^2$ is guaranteed to flow into the attractor corresponding to \mathbf{s}^* . Third, there are no limit cycles. Fourth, for satisfiable formulae the only fixed point attractors of the dynamics are the global minima of V with $V = 0$. Note that in principle, the projection of the dynamics onto \mathcal{H}_N could be stuck in some point \mathbf{s} , while $da/dt \neq 0$ indefinitely. This does not

¹Departments of Physics, University of Notre Dame, Notre Dame, Indiana 46556, USA, ²Interdisciplinary Centre for Network Science and Applications (iCeNSA), University of Notre Dame, Notre Dame, Indiana 46556, USA, ³Departments of Computer Science and Engineering, University of Notre Dame, Notre Dame, Indiana 46556, USA. *e-mail: mercseyr@nd.edu; toro@nd.edu.

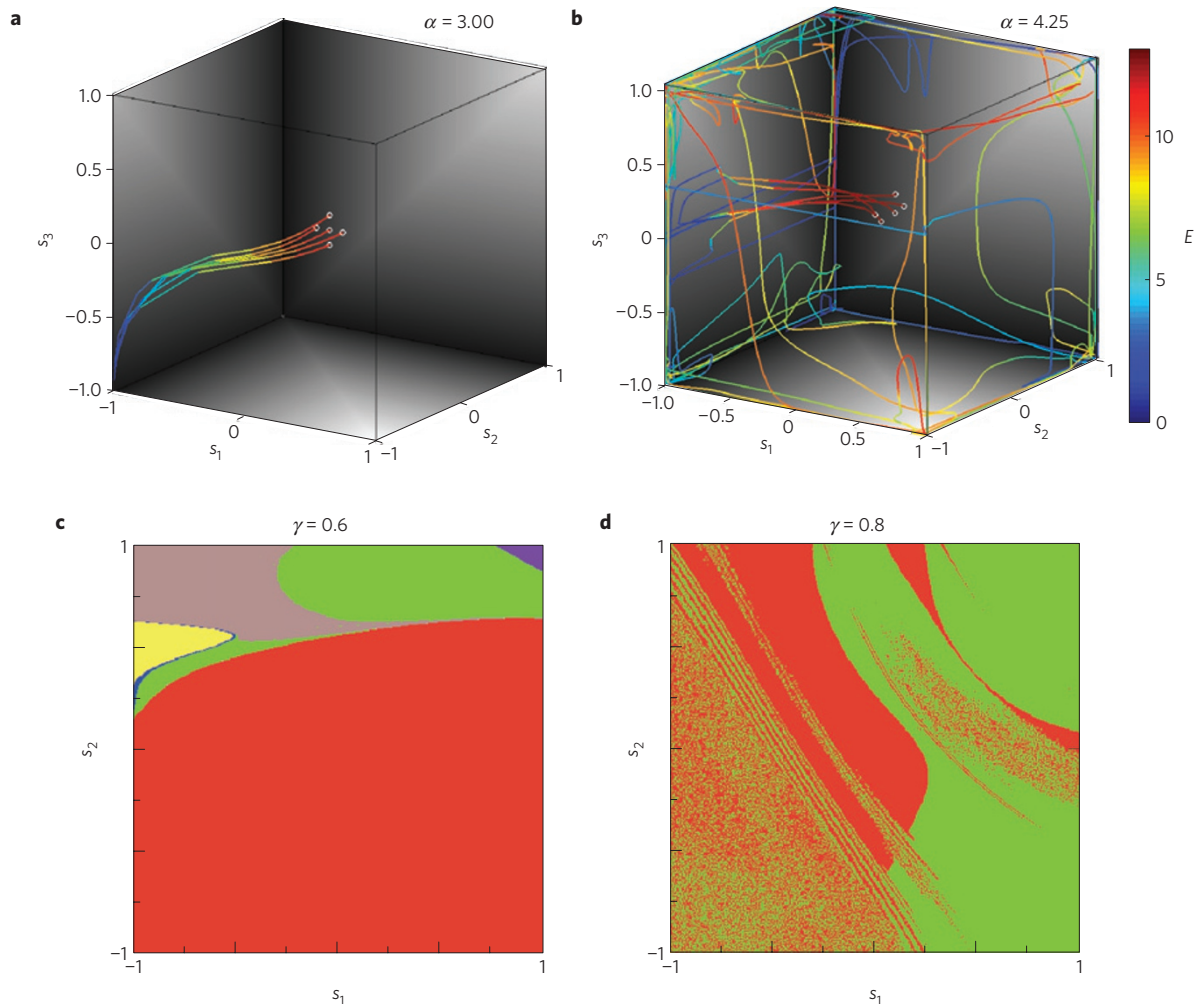


Figure 1 | Chaotic behaviour. **a,b**, Five, closely started sample trajectories projected onto (s_1, s_2, s_3) , for a 3-SAT formula with $N = 200$, $\alpha = 3$ (**a**) and for a hard formula, $N = 200$, $\alpha = 4.25$ (**b**). The colour indicates the energy E (colour bar) in a given point of the trajectory. Whereas for easy formulae the trajectories exhibit laminar flow, for hard formulae they quickly become separated, showing a chaotic evolution. **c,d**, Taking a small 3-XORSAT instance with $N = 15$ (see Supplementary Section SG) we fix a random initial condition for all s_i , except s_1 and s_2 , which are varied on a 400×400 grid, and we colour each point according to the solution they flow to for $\gamma = 0.6$ (**c**; instance shown in Supplementary Fig. S4e) and for $\gamma = 0.8$ (**d**; instance shown in Supplementary Fig. S4f).

happen here, as shown in Supplementary Section SE. Moreover, analytical arguments supported by simulations indicate that the trajectory will leave any domain that does not contain solutions, see the discussion in Supplementary Section SE1. Note that the constraint functions (hence their satisfiability) depend directly only on the location of the trajectory in \mathcal{H}_N , $K_m = K_m(\mathbf{s})$, and not on the auxiliary variables. The dynamics in the \mathbf{a} -space is simple expansion, and for this reason the features of the full phase space Ω lie within its projection onto \mathcal{H}_N . One can actually eliminate entirely the auxiliary variables from the equations by first solving (1b) to give $a_m(t) = a_m(0) \exp(\int_0^t K_m(\mathbf{s}(\tau)) d\tau)$ and then inserting it into (1a).

Another fundamental feature of (1) is that it is deterministic: for a given formula f , any initial condition generates a unique trajectory, and any set from \mathcal{H}_N has a unique pre-image arbitrarily back in time. Hence, the characteristics of the solution space are reflected in the properties of the invariant sets⁷ of the dynamics (1) within the hypercube \mathcal{H}_N . The deterministic nature of (1) allows us to define basins of attractions of solution clusters by colouring every point in \mathcal{H}_N according to which cluster the trajectory flows to, if started from there. These basins fill \mathcal{H}_N up to a set of zero (Lebesgue) measure, which forms the basin boundary⁷, from where the dynamics (by definition) cannot flow to any of the

attractors. A k -SAT formula f can be represented as a hypergraph $\mathcal{G}(f)$ (or equivalently, a factor graph) in which nodes are variables and hyperedges are clauses connecting the nodes/variables in the clause. Pure literals are those that participate in one or more clauses but always in the same form (direct or negated); hence, they can always be chosen such as to satisfy those clauses. The core of $\mathcal{G}(f)$ is the subgraph left after sequentially removing all of the hyperedges having pure literals²³. For simple formulae (such as those without a core), the dynamics of (1) is laminar flow and the basin boundaries form smooth, non-fractal sets (Figs 1a,c and 2, top two rows). Adding more constraints $\mathcal{G}(f)$ develops a core, the spin equations (1a) become mutually coupled, and the trajectories may become chaotic (Fig. 1b, Supplementary Section SF and Fig. S8) and the basin boundaries fractal⁷⁻⁹ (Figs 1d, 2 and Supplementary Fig. S4). Therefore, as the constraint density $\alpha = M/N$ is increased within predefined ensembles of formulae (random k -SAT, occupation problems, k -XORSAT and so on) a sharp change to chaotic behaviour is expected at a chaotic transition point α_c , where a chaotic core appears with non-zero statistical weight in the ensemble as $N \rightarrow \infty$. As an example, let us consider 3-XORSAT. In this case, owing to its inherently linear nature, it is actually better to work directly with the parity check equations as

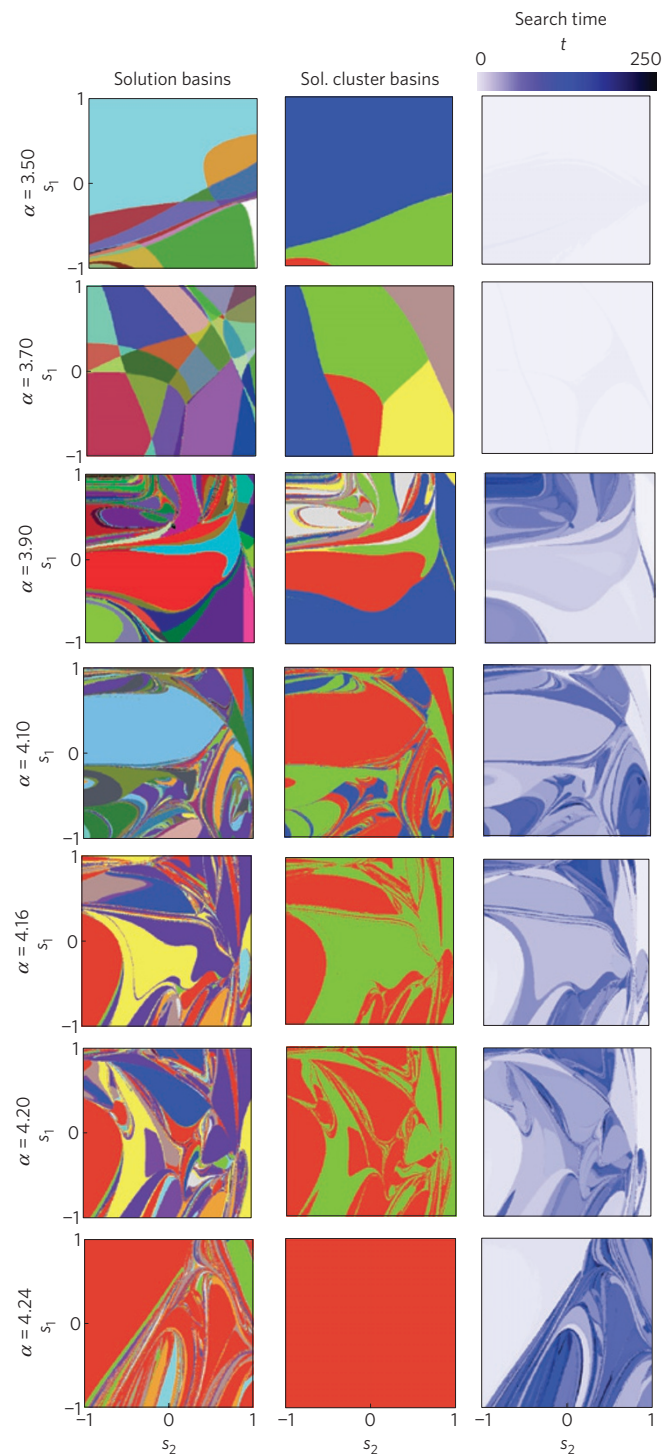


Figure 2 | Attractor basins. For a random 3-SAT instance with $N = 50$ we vary α by successively adding new constraints. Fixing a random initial condition for $s_i, i \geq 3$, we vary only s_1 and s_2 on a 400×400 grid, and we colour each point according to the solution (first column) or solution cluster (second column) they flow to. Each colour in a given column represents a solution or solution cluster respectively; however, colours between columns are independent. The third column represents the analog search time t needed to find a solution (see colour bar) starting from the corresponding grid point. Maps are presented for values of $\alpha = 3.5, 3.7, 3.9, 4.1, 4.16, 4.2$ and 4.24 . Easy formulae are characterized by smooth basin boundaries and small search times. Note that we see only the solutions (and clusters) that reveal themselves in the (s_1, s_2) plane; others might not be seen. For hard formulae the boundaries and the search time maps become fractal.

constraints, instead of their conjunctive normal form. The chaotic core here is a small finite hypergraph, and thus α_χ coincides with the so-called dynamical transition point α_d computed exactly in ref. 24 (see Supplementary Section SG and Fig. S4). Note, a core can be non-chaotic, and thus the existence of a core is only a necessary condition for the appearance of chaos and in general the two transitions might not coincide. Further increasing the number of constraints (within any formula ensemble) unsatisfiability appears at the threshold value $\alpha_s > \alpha_\chi$ beyond which almost all formulae are unsatisfiable (UNSAT regime)^{11,12,22,24–28}. The closer α is to α_s , the harder it is to find solutions, and beyond the so-called freezing transition point $\alpha_f < \alpha_s$ (called the frozen regime) all known algorithms take exponentially long times or simply fail to find solutions^{11,12}. A variable is frozen if it takes on the same value for all solutions within a cluster, and a cluster is frozen if an extensive number of its variables are frozen. In the frozen regime all clusters are frozen and they are also far apart ($\mathcal{O}(N)$ Hamming distance)^{11,12}. For random 3-SAT (clauses chosen uniformly at random for fixed α) $\alpha_s \cong 4.26$ (ref. 27), $\alpha_f \cong 4.25$ (ref. 28) and all known local search algorithms become exponential or fail beyond $\alpha = 4.21$ (ref. 29), and survey-propagation²⁵-based algorithms fail beyond $\alpha = 4.25$ (ref. 28). As the frozen regime is very thin in random 3-SAT, the so-called locked occupation problems (LOPs) have been introduced¹². In LOPs all clusters are formed by exactly one solution; hence, they are completely frozen and the frozen regime extends from the clustering (dynamical) transition point ℓ_d to the satisfiability threshold ℓ_s , and thus it is very wide¹². An example LOP is random ‘+1-in-3-SAT’ (ref. 12), made of constraints that have no negated variables and a constraint is satisfied only if exactly one of its variables is 1 (TRUE). In +1-in-3-SAT $\ell_d \cong 2.256$, $\ell_s \cong 2.368$ and beyond ℓ_d all known algorithms have exponential search times or fail to find solutions (here $\ell = 3M/N$).

As chaos is present for satisfiable formulae, that is, when system (1) has attracting fixed points, it is necessarily of transient type. Transient chaos^{4–7} is ubiquitous in systems with many degrees of freedom such as fluid turbulence³⁰. It appears as the result of homoclinic/heteroclinic intersections of the invariant manifolds of hyperbolic (unstable) fixed points of (1) lying within the basin boundary^{7–9}, leading to complex (fractal) foliations of the phase space (see Supplementary Section SF). We observed the prevalence of transient chaos in the whole region $\alpha_\chi < \alpha < \alpha_s$ for all of the problem classes we studied. Interestingly, the velocity fluctuations of trajectories in the chaotic regime are qualitatively similar to those of fluid parcels in turbulent flows as shown in Supplementary Section SK. Our findings indicate that chaotic behaviour may be a generic feature of algorithms searching for solutions in hard optimization problems, corroborating previous observations¹⁰ using a heuristic algorithm based on iterated maps.

In the following we show results on random 3-SAT and +1-in-3-SAT formulae in the frozen regime; however, the same conclusions hold for other ensembles that we tested. To investigate the complexity of computation by the flow (1), we monitored the fraction of problems $p(t)$ not solved by continuous time t , as a function of N and α . Figure 3a,c shows that even in the frozen phase, the fraction of unsolved problems by time t decays exponentially with t , that is, by a law $p(t) = re^{-\lambda(N)t}$. The decay rate $\lambda(N)$ obeys $\lambda(N) = bN^{-\beta}$, with $\beta \approx 1.6$ in both cases, see Fig. 3b,d. From these two equations, the continuous time $t(p, N)$ needed to solve a fixed $(1 - p)$ th fraction of random formulae (or to miss solving the p th fraction of them) is:

$$t(p, N) = b^{-1} N^\beta \ln(r/p) \quad (2)$$

indicating that the continuous time needed to find solutions scales as a power law with N . Equation (2) also implies power-law scaling for almost all hard instances in the $N \rightarrow \infty$ limit (Supplementary

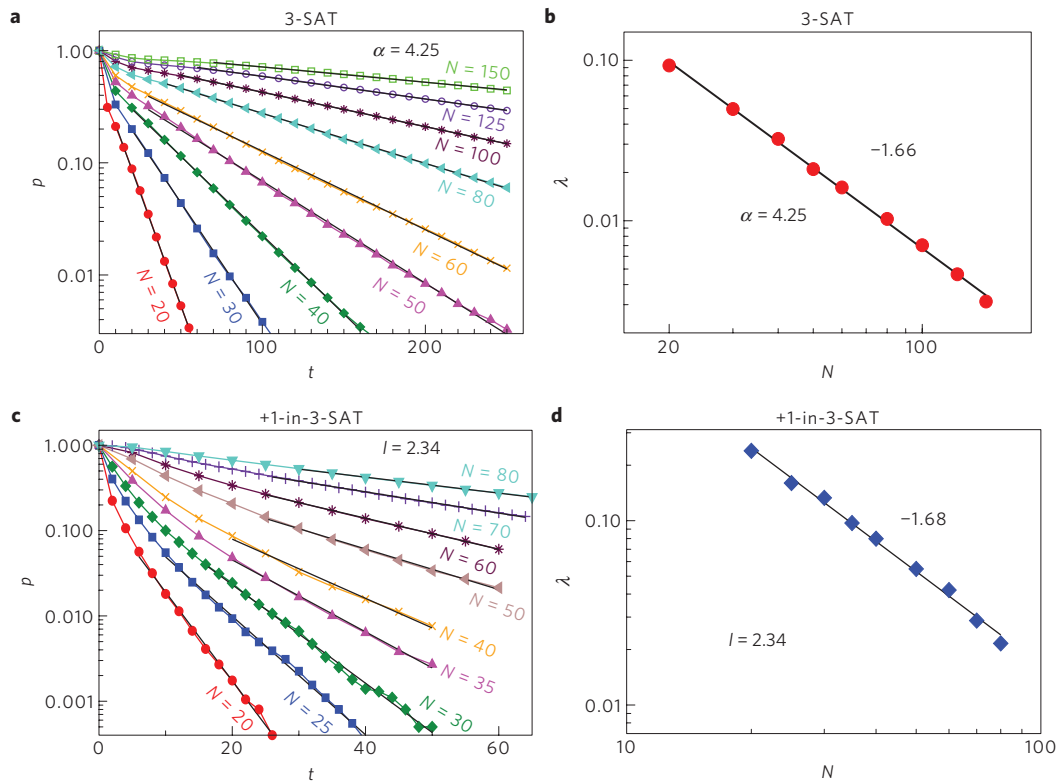


Figure 3 | Computational complexity properties. **a**, The fraction of problems $p(t)$ not yet solved by continuous time t for 3-SAT at $\alpha = 4.25$, for $N = 20, 30, 40, 50, 60, 80, 100, 125$ and 150 (colours). Averages were done over 10^5 instances for each N . For each instance the dynamics was started from one random initial condition. Black continuous lines show the decay $p(t) = \text{rexp}(-\lambda(N)t)$. **b**, The decay rate follows $\lambda(N) = bN^{-\beta}$, with $\beta \approx 1.66$. **c**, The fraction of problems $p(t)$ unsolved by time t for +1-in-3-SAT at $l = 2.34$, for $N = 20, 25, 30, 35, 40, 50, 60, 70$ and 80 . For each instance the dynamics was started in parallel from 10 random initial conditions; averages were taken over 10^4 instances for each N . Black continuous lines show the same exponential decay as in **a**. **d**, The decay rate shows the same behaviour as in **b** with exponent $\beta \approx 1.68$.

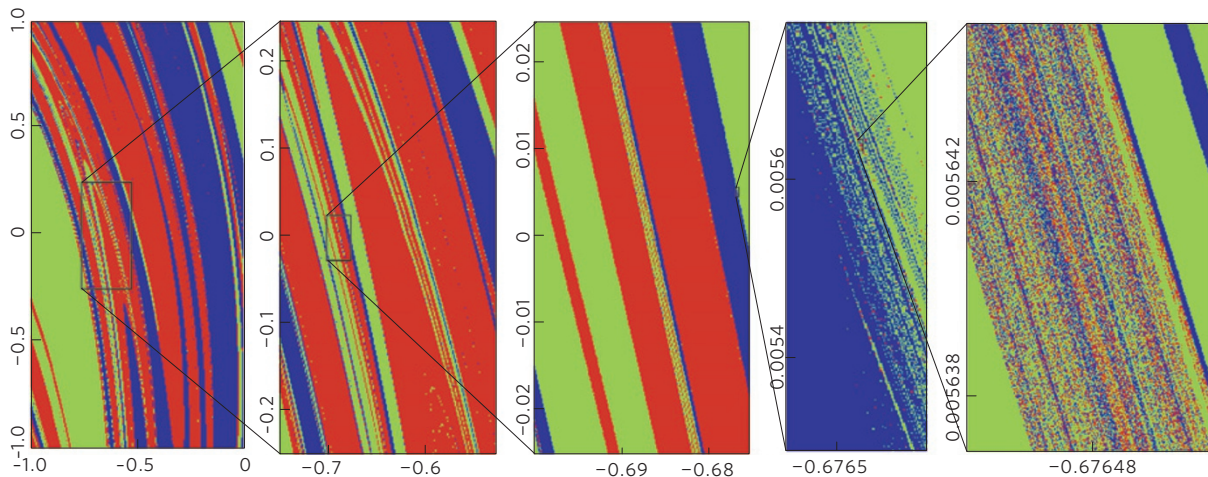


Figure 4 | Wada property. Basin boundaries are shown for +1-in-3-SAT for an instance at $N = 30$ and $l = 2.28$. Fixing a random initial condition for all $s_i, i \geq 3$ we vary only s_1 and s_2 on a 200×400 grid and colour the points according to three different solutions they flow to. Successive magnifications illustrate the Wada property: the points on the basin boundaries are simultaneously on the boundary of all three basins, implying that large enough magnifications will contain all three colours (although the blue-green boundary seems void of red in the third panel, panels four and five show that red is actually present).

Section SH). The length in \mathcal{H}_N of the corresponding continuous trajectories also scales as a power law with N (Supplementary Fig. S7b and Section SJ). However, note that this does not mean that the algorithm itself is a polynomial-cost algorithm, as the energy function V can have exponentially large fluctuations. As the numerical integration happens on a digital machine, it approximates the continuous trajectory with discrete points.

Monitoring the fraction of formulae left unsolved as a function of the number of discretization steps n_{step} in the frozen phase, we find exponential behaviour for $n_{\text{step}}(p, N)$ (Supplementary Sections SI, SJ and Fig. S6). The difference between the continuous- and discrete-time complexities is due to the wildly fluctuating nature of the chaotic trajectories (see Fig. 1b and Methods) in the frozen phase. Compounding this, we also observe the appearance of the

Wada property^{7,8} in the basin boundaries (Fig. 4). A fractal basin boundary has Wada property if its points are simultaneously on the boundary of at least three colours/basins. (An amusing method that creates such sets uses four Christmas ball ornaments⁷.) Although the Wada property does not affect the true/mathematical analog trajectories, owing to numerical errors, it may switch the numerical trajectories between the basins. As the clusters are far ($\mathcal{O}(N)$) apart, the switched trajectory will flow towards another cluster into a practically opposing region of \mathcal{H}_N until it may come close again to the basin boundary and so on, partially randomizing the trajectory in \mathcal{H}_N .

We conjecture that the power-law scaling of the continuous search times (2) is due in part to a generic property of the dynamical system (1), namely that it is hyperbolic^{4,6,13} or near-hyperbolic. It has been shown that for hyperbolic systems the trajectories escape from regions far away from the attractors to the attractors at an exponential rate, for almost all initial conditions^{4,6,13}. That is, the fraction of trajectories still searching for a solution after time t decays as $e^{-\kappa t}$ (Supplementary Fig. S9), where κ is the escape rate. Thus, κ^{-1} can be considered as a measure of hardness for a given formula. When taken over an ensemble at a given α , this property generates the exponential decay for $p(t)$ with an average escape rate λ .

The form of the energy function V incorporates the influence of all of the clauses at all times, and in this sense (1) is a non-local search algorithm. As shown before, the auxiliary variables can be eliminated; however, they give a convenient interpretation of the dynamics. Namely, one can think of them as providing extra dimensions along which the trajectories escape from local wells, and their form (1b) provides positive feedback that guarantees their escape. Clearly, these equations are not unique, and other forms based on the same principles may work just as well.

Methods

To simulate (1), we use a fifth-order adaptive Cash–Karp Runge–Kutta method with monitoring of local truncation error to ensure accuracy. To keep the numerical trajectory within a tube of small, preset thickness around the true analog trajectory in Ω (Supplementary Fig. S5), the Runge–Kutta algorithm occasionally carries out an exponentially large number of discretization steps n_{step} . However, this happens only for hard formulae, when the analog trajectory has wild, chaotic fluctuations. For easy formulae, both $p(t)$ and $p(n_{\text{step}})$ decay exponentially as shown in Supplementary Fig. S6a, inset.

Received 16 September 2010; accepted 18 August 2011;
published online 2 October 2011

References

1. Cook, S. in *Proc. Third Ann. Symp. Theory of Computing* 151–158 (ACM, 1971).
2. Fortnow, L. The status of the P versus NP problem. *Commun. ACM* **52**, 78–86 (2009).
3. Garey, M. R. & Johnson, D. S. *Computers and Intractability: A Guide to the Theory of NP-Completeness* (W. H. Freeman, 1990).
4. Kadanoff, L. P. & Tang, C. Escape from strange repellers. *Proc. Natl Acad. Sci.* **81**, 1276–1279 (1984).
5. Tél, T. & Lai, Y.-C. Chaotic transients in spatially extended systems. *Phys. Rep.* **460**, 245–275 (2008).
6. Lai, Y.-C. & Tél, T. *Transient Chaos: Complex Dynamics on Finite-Time Scales* (Springer, 2011).
7. Ott, E. *Chaos in Dynamical Systems* 2nd edn (Cambridge Univ. Press, 2002).

8. Nusse, H. E. & Yorke, J. A. Basins of attraction. *Science* **271**, 1376–1380 (1996).
9. Grebogi, C., Ott, E. & Yorke, J. A. Basin boundary metamorphoses: Changes in accessible boundary orbits. *Physica D* **24**, 243–262 (1987).
10. Elser, V., Rankenburg, I. & Thibault, P. Searching with iterated maps. *Proc. Natl Acad. Sci.* **104**, 418–423 (2007).
11. Achlioptas, D. & Ricci-Tersenghi, F. Random formulas have frozen variables. *SIAM J. Comput.* **39**, 260–280 (2009).
12. Zdeborová, L. & Mézard, M. Locked constraint satisfaction problems. *Phys. Rev. Lett.* **101**, 078702 (2008).
13. Cvitanović, P., Artuso, R., Maineri, R., Tanner, G. & Vattay, G. *Chaos: Classical and Quantum* (Niels Bohr Institute, 2010) ChaosBook.org/version13.
14. Branicky, M. *IEEE Workshop on Physics and Computation* 265–274 (IEEE Computer Society Press, 1994).
15. Siegelmann, H. T. Computation beyond the Turing limit. *Science* **268**, 545–548 (1995).
16. Moore, C. Recursion theory on the reals and continuous-time computation. *Theor. Comput. Sci.* **162**, 23–44 (1996).
17. Liu, S.-C., Kramer, J., Indiveri, G., Delbruck, T. & Douglas, R. *Analog VLSI: Circuits and Principles* (MIT Press, 2002).
18. Chua, L. O. & Roska, T. *Cellular Neural Networks and Visual Computing: Foundations and Applications* (Cambridge Univ. Press, 2005).
19. Gu, J., Gu, Q. & Du, D. On optimizing the satisfiability (SAT) problem. *J. Comput. Sci. Technol.* **14**, 1–17 (1999).
20. Nagamatu, M. & Yanaru, T. On the stability of Lagrange programming networks for satisfiability problems of propositional calculus. *Neurocomputing* **13**, 119–133 (1996).
21. Wah, B. W. & Chang, Y.-J. Trace-based methods for solving nonlinear global optimization and satisfiability problems. *J. Glob. Opt.* **10**, 107–141 (1997).
22. Achlioptas, D., Coja-Oghlan, A. & Ricci-Tersenghi, F. On the solution-space geometry of constraint satisfaction problems. *Random Struct. Algorithms* **38**, 251–268 (2011).
23. Molloy, M. Cores in random hypergraphs and Boolean formulas. *Random Struct. Algorithms* **27**, 124–135 (2005).
24. Mézard, M., Ricci-Tersenghi, F. & Zecchina, R. Two solutions to diluted p -spin models and XORSAT problems. *J. Stat. Phys.* **111**, 505–533 (2003).
25. Mézard, M., Parisi, G. & Zecchina, R. Analytic and algorithmic solution of random satisfiability problems. *Science* **297**, 812–815 (2002).
26. Achlioptas, D., Naor, A. & Peres, Y. Rigorous location of phase transitions in hard optimization problems. *Nature* **435**, 759–764 (2005).
27. Mertens, S., Mézard, M. & Zecchina, R. Threshold values of random k -SAT from the cavity method. *Random Struct. Algorithms* **28**, 340–373 (2006).
28. Parisi, G. Some remarks on the survey decimation algorithm for k -satisfiability. <http://arxiv.org/abs/cs/0301015> (2003).
29. Seitz, S., Alava, M. & Orponen, P. Focused local search for random 3-satisfiability. *J. Stat. Mech.* P06006 (2005).
30. Hof, B., de Lozar, A., Kuik, D.J. & Westerweel, J. Repeller or Attractor? Selecting the dynamical model for the onset of turbulence in pipe flow. *Phys. Rev. Lett.* **101**, 214501 (2008).

Acknowledgements

We thank T. Tél and L. Lovász for valuable discussions and for a critical reading of the manuscript.

Author contributions

M.E.-R. and Z.T. conceived and designed the research and contributed analysis tools equally. M.E.-R. carried out all simulations, collected and analysed all of the data and Z.T. wrote the paper.

Additional information

The authors declare no competing financial interests. Supplementary information accompanies this paper on www.nature.com/naturephysics. Reprints and permissions information is available online at <http://www.nature.com/reprints>. Correspondence and requests for materials should be addressed to M.E.-R. or Z.T.