Journal of Statistical Mechanics: Theory and Experiment
An IOP and SISSA journal

On the cavity method for decimated random constraint satisfaction problems and the analysis of belief propagation guided decimation algorithms

Federico Ricci-Tersenghi[1] **and Guilhem Semerjian**[2]

[1] Dipartimento di Fisica, Università di Roma La Sapienza, P. A. Moro 2, I-00185 Roma, Italy
[2] LPTENS, Unité Mixte de Recherche (UMR 8549) du CNRS et de l'ENS associée à l'université Pierre et Marie Curie, 24 Rue Lhomond, F-75231 Paris Cedex 05, France
E-mail: Federico.Ricci@roma1.infn.it and guilhem.semerjian@lpt.ens.fr

Received 10 June 2009
Accepted 1 August 2009
Published 9 September 2009

Online at stacks.iop.org/JSTAT/2009/P09001
doi:10.1088/1742-5468/2009/09/P09001

Abstract. We introduce a version of the cavity method for diluted mean-field spin models that allows the computation of thermodynamic quantities similar to the Franz–Parisi quenched potential in sparse random graph models. This method is developed in the particular case of partially decimated random constraint satisfaction problems. This allows us to develop a theoretical understanding of a class of algorithms for solving constraint satisfaction problems, in which elementary degrees of freedom are sequentially assigned according to the results of a message passing procedure (belief propagation). We confront this theoretical analysis with the results of extensive numerical simulations.

Keywords: cavity and replica method, analysis of algorithms, message-passing algorithms

ArXiv ePrint: 0904.3395

1742-5468/09/P09001+46$30.00

Contents

J. Stat. Mech. (2009) P09001

1. Introduction

In constraint satisfaction problems (CSP) a set of variables is required to simultaneously satisfy a series of constraints. One can equivalently define an energy function as the number of unsatisfied constraints of a given assignment of the variables, and rephrase the CSP as the quest for a zero-energy ground state configuration. This analogy with low temperature physics triggered an intensive research effort within the statistical mechanics community. More precisely, one line of approach for these problems consists in looking for typical properties of randomly generated large instances. This translates into the presence of quenched disorder in the corresponding physical model. The constraints to be satisfied are generally contradicting each other, and the definition of the random instances does not involve an underlying finite-dimensional space; as a consequence these problems fall into the category of mean-field spin glasses, for which a set of analytical tools have been developed during the last decades [1]–[3].

The most famous example of random CSP is the random k-sat ensemble. Statistical mechanics studies have led to two kind of results for this problem. On the one hand, qualitative and quantitative predictions have been made about the various phase transitions encountered for the typical behavior of large instances when the control parameter governing the amount of frustration is varied. The satisfiability transition marks the sudden disappearance of the solutions (zero-energy ground states). There exist rigorous results on the properties of this transition [4] and bounds [5, 6] on its possible location. Statistical mechanics has complemented these results with a heuristic way to compute this threshold, hence yielding quantitative conjectures on its value [7]–[9]. Another important contribution has been the suggestion of other phase transitions in the satisfiable regime, which concerns the geometrical organization of the set of solutions inside the configuration space [7, 8, 10, 11]. For large enough frustration, but below the satisfiability transition, the solutions can be grouped in clusters of nearby solutions, each cluster being separated from the others.

On the other hand, attention has also been paid to algorithmic issues, that is to procedures aiming at solving CSP, by finding their solutions or proving that no solution exists. These algorithms can be roughly divided in two broad categories: local search and sequential assignment procedures. In the first one, which has also been studied with statistical mechanics methods [12]–[14], a random walk is performed in the configuration space, with transition rules tuned to bias the walk towards the solutions. This kind of algorithm is called incomplete: it cannot prove the absence of a solution if it fails to find one. The second category proceeds differently: at some step of the algorithm only one part of the variables has a definite value, the others being still free. Each step thus corresponds to the choice of one free variable and of the value it will be assigned to, the CSP on the remaining variables being consequently simplified. The heuristics guiding these choices can be more or less elaborate. In the simplest cases one only takes into account simple properties of the free variables, such as the number of their occurrences in the remaining CSP. A rigorous analysis of these simple ('myopic') approaches is possible and is at the basis of most of the lower bounds on the satisfiability transition [5]. Such algorithms can be made complete if backtracking is allowed, i.e. choices which have led to a contradiction can be corrected in a systematic way. This complete version of the procedure is called the DPLL algorithm [15].

doi:10.1088/1742-5468/2009/09/P09001

J. Stat. Mech. (2009) P09001

One outcome of the statistical mechanics studies of random CSP has been the proposal of an incomplete sequential assignment algorithm called survey propagation-inspired decimation [7, 16], which proved to be very efficient on satisfiable random instances close to the satisfiability transition. This algorithm relies on the clustering picture of the solution space in the satisfiable regime. Unfortunately its theoretical analysis is much more difficult than for myopic ones. Indeed the heuristics of choice of the variables to be assigned is based on the result of a message-passing iterative computation which depends on the whole remaining CSP in an intricate way. More generally, the analysis of message-passing decimation procedures is difficult and there are few results on this issue, with the notable exception of [17]–[19] for the so-called warning propagation algorithm on overconstrained satisfiability formulae.

In this paper we provide an analytical description of an algorithm similar to survey propagation, yet simpler. It has been studied numerically in [20, 21]. A part of our results were published in [22]; the method developed there was also applied to another family of CSP in [23]. In the sequential assignment procedure under investigation the choice of the value of the assigned variable is made at each step according to the belief propagation message-passing algorithm (instead of survey propagation). it aims at mimicking the following ideal procedure. After a certain number of variables has been assigned, one can define the uniform probability measure over the solutions of the CSP which are still compatible with the previous choices. If one were able to compute the marginal probabilities of this (conditional) probability measure and use them to draw the value of the newly assigned variable at each step, one would construct a uniform sampler of the solutions of the original CSP, and this would in particular lead to an algorithm for finding one solution of the CSP. The computation of these marginal probabilities is a computationally intractable task; belief propagation is a fast heuristic algorithm, widely used for inference problems [24, 25], which is often able to compute good approximations of these marginal probabilities. Analyzing the behavior of the belief-inspired decimation procedure thus amounts to controlling the error which accumulates at each step by using the BP approximate estimates of the marginal probabilities instead of the exact ones. A theoretical understanding and quantitative description of the deviations between exact and BP-computed marginal probabilities for graphical models is a formidable open problem that we shall not attack directly in this paper. We will instead perform a theoretical analysis of the putative algorithm based on an hypothetical exact marginal computation. This analysis will be obtained by a generalization of the cavity method which is able to deal with the partially decimated CSP encountered along the execution of the algorithm, and to compute the extended phase diagram of these problems. This approach is technically similar to the computation of Franz–Parisi quenched potentials [26]. The relevance of this theoretical analysis for the understanding of the approximate BP implementation will then be argued on the basis of a comparison with extensive numerical simulations.

This paper is organized as follows. In section 2 we give a more precise definition of the ideal decimation procedure sketched above and explain how an approximate realization of this idea can be performed in practice. Section 3 is devoted to the cavity method for decimated formulae that provides an analytical description of the ideal decimation procedure. In the next two sections we apply this formalism to two specific CSP, and compare its predictions to the results of numerical simulations of the BP guided decimation algorithm. We begin in section 4 with the xor-satisfiability problem, a well-studied simple

J. Stat. Mech. (2009) P09001

example for which many results can be checked with alternative techniques. We then turn to the case of k-satisfiability random instances in section 5. We draw our conclusions in section 6. More technical details are deferred to the appendix.

2. A thought experiment

2.1. Definition of random CSPs and a brief review of their properties

A constraint satisfaction problem (CSP) is defined on a set of N variables $\sigma_1, \dots, \sigma_N$, taking values in a finite alphabet. We shall denote $\underline{\sigma} = (\sigma_1, \dots, \sigma_N)$ the global configuration of the variables, and for a subset S of the indices $\{1, \dots, N\}$ we call $\underline{\sigma}_S = \{\sigma_i, i \in S\}$ the partial configuration of the variables in S. The solutions of the CSP are the configurations which simultaneously satisfy M constraints (also called clauses in the following), each of them being specified by a function $\psi_a(\underline{\sigma}_{\partial a})$ of a subset ∂a of the variables. The function ψ_a takes value 1 (resp. 0) whenever the constraint is satisfied (resp. unsatisfied). A CSP admits a natural representation in terms of a factor graph [24], i.e. a bipartite graph where one type of vertex (variable node) is associated to each variable $i = 1, \dots, N$ and another type (function node) to each constraint $a = 1, \dots, M$. An edge links the ith variable node with the ath function node whenever the constraint ψ_a depends on σ_i, i.e. in the notation introduced above whenever $i \in \partial a$. We shall similarly denote ∂i the set of function nodes which depend on the ith variable, and define the distance between two variable nodes i and j as the minimal number of constraint nodes encountered on a path of the factor graph joining i and j.

In the following we will concentrate on two examples of CSP, both on binary variables that we shall represent by Ising spins, $\sigma_i = \pm 1$:

- k-xorsat. Each constraint a depends on k distinct variables $\partial a = \{i_1^a, \dots, i_k^a\}$, and requires the product of the corresponding spins to take a given value $J^a \in \{-1, +1\}$:

$$\psi_a(\underline{\sigma}_{\partial a}) = \mathbb{I}\left(\prod_{i \in \partial a} \sigma_i = J^a\right), \tag{1}$$

  where here and in the following $\mathbb{I}(\cdot)$ is the indicator function of an event. This condition is easily seen to be equivalent to a constraint on the value of the eXclusive OR of k Boolean variables, hence the name of the problem.
- k-sat. The constraint a depends again on $\partial a = \{i_1^a, \dots, i_k^a\}$, but imposes the configuration of these k variables to avoid one out of the 2^k possible ones:

$$\psi_a(\underline{\sigma}_{\partial a}) = 1 - \mathbb{I}(\sigma_i = J_i^a \ \forall i \in \partial a), \tag{2}$$

  where $(J_{i_1^a}^a, \dots, J_{i_k^a}^a) \in \{-1, +1\}^k$ are fixed constants defining the constraint. Equivalently, one requires the logical OR of k literals (a Boolean variable or its negation) to evaluate to TRUE.

From a computational complexity point of view these two problems are very different. The decision version of a CSP consists in determining whether it admits at least one solution, i.e. one configuration satisfying all constraints simultaneously. The k-xorsat decision problem belongs to the easy, polynomial P complexity class [27] for any value of

J. Stat. Mech. (2009) P09001

k. One can indeed use Gaussian elimination to check if the associated system of linear equations modulo 2 is solvable. k-sat is, in contrast, NP-complete for all $k \geq 3$: no algorithm able to decide the satisfiability of every k-sat formula in a time bounded by a polynomial of the formula size is known.

Despite this deep difference in the worst-case point of view, these two families of problems share common features in their 'average complexity' behavior. By this we mean the random ensembles of instances that have been extensively studied in the computer science and statistical physics literature and that are defined as follows. A random k-xorsat formula is generated by drawing in an independent, identical way M constraints; the k-uplet of indices ∂a is drawn uniformly among the $\binom{N}{k}$ possible ones, and the coupling constant J^a is taken to be ± 1 with equal probability of one-half. The generation of a random k-sat formula is similar, with for all constraints a the k constants $\{J_i^a\}_{i\in\partial a}$ being taken independently equal to ± 1 with equal probability. These random ensembles exhibit a rich phenomenology in the thermodynamic limit $N, M \to \infty$ with $\alpha = M/N$ fixed. In particular a satisfiability phase transition occurs at a value $\alpha_{\rm s}$ (which depends on the value of k and on the problem, sat or xorsat, under consideration): random formulae with $\alpha < \alpha_{\rm s}$ are, with high probability, satisfiable, whereas for $\alpha > \alpha_{\rm s}$ they are unsatisfiable. Here and in the following 'with high probability' (w.h.p.) means with a probability going to one in the above stated thermodynamic limit. To be more precise, for xorsat this statement has been proven and the values of $\alpha_{\rm s}$ have been computed [28, 29]. For sat random formulae this satisfiability transition is, strictly speaking, only a conjecture. The existence of a tight threshold $\alpha_{\rm s}(N)$ has been proven in [4], but not the convergence of $\alpha_{\rm s}(N)$, that could in principle oscillate between the bounds on its possible location [5, 6]. It is, however, most probable that the values of $\alpha_{\rm s}$ computed within the statistical mechanics framework [7]–[9] are exact.

Besides the satisfiability transition other interesting phenomena occur in the satisfiable phase $\alpha < \alpha_{\rm s}$. In this regime the formulas are w.h.p. satisfiable and in fact admit an exponential number of solutions; however, there are structural phase transitions at which the properties of the set of solutions change qualitatively. To describe the set of solutions it is convenient to introduce the uniform probability measure on this set:

$$\mu(\underline{\sigma}) = \frac{1}{Z}\prod_{a=1}^{M}\psi_a(\underline{\sigma}_{\partial a}), \tag{3}$$

where the normalizing factor Z is the number of solutions of the CSP. Note that this probability measure is itself a random object, as it depends on the instance of the CSP under study, and that it is defined only for the satisfiable instances, which is the case w.h.p. in the regime $\alpha < \alpha_{\rm s}$ we are considering here.

In the xorsat ensemble of random formulae there is a single structural phase transition in the satisfiable regime [28, 29], known as the clustering transition with a threshold denoted $\alpha_{\rm d}$. For lower values of the connectivity α, the set of solutions is well connected. In contrast, for $\alpha_{\rm d} < \alpha < \alpha_{\rm s}$, the exponential number of solutions gets split into an exponential number $\exp[N\Sigma]$ of clusters, separated from each other in the configuration space. The rate of growth Σ of the number of clusters is usually called the complexity, it decreases when α grows and vanishes at the satisfiability threshold. In the clustered phase each cluster contains the same exponential number of solutions (with a rate of

J. Stat. Mech. (2009) P09001

growth called internal entropy). The structure of xorsat is sufficiently simple for a clear-cut definition of the clusters to be possible. In fact the clustering transition corresponds to a percolation transition in the associated factor graph, where an extensive two-core discontinuously appears.

The structure of the satisfiable phase of the random satisfiability ensemble is richer [11, 30]. At the clustering transition [7, 10] the exponentially numerous clusters have, contrary to xorsat, a large diversity of internal entropies. This leads, for $k \geq 4$, to another transition, the condensation one at α_c. When $\alpha_d < \alpha < \alpha_c$ the measure (3) is split into an exponential number of clusters, while for $\alpha_c < \alpha < \alpha_s$ almost all solutions are contained in a sub-exponential number of clusters. These various phase transitions can be characterized in terms of the strength of the correlations between variables. The clustering α_d is related to the appearance of long-range point-to-set correlations, or in other words to the possibility of reconstruction of the value of a variable given the values of all the variables at a large distance from it [31]. At the condensation transition non-trivial correlations are already revealed by the correlation functions between a finite number of variables [11].

2.2. Oracle-guided algorithm and ensemble of decimated CSPs

The study presented in this paper is based on the analysis of an ideal procedure to find the solutions of a CSP, which we discuss here more precisely. Consider a satisfiable CSP instance, and the uniform measure $\mu(\cdot)$ over its solutions defined in (3). Let us also introduce a subset D of the variables, and a partial configuration of these variables $\underline{\tau}_D$ compatible with at least one solution of the instance. We can thus define a conditional version of μ, $\mu(\cdot|\underline{\tau}_D)$, which is the uniform measure over the solutions of the formula compatible with $\underline{\tau}_D$:

$$\mu(\underline{\sigma}|\underline{\tau}_D) = \begin{cases} \dfrac{1}{Z(\underline{\tau}_D)} \displaystyle\prod_{a=1}^{M} \psi_a(\underline{\sigma}_{\partial a}) & \text{if } \underline{\sigma}_D = \underline{\tau}_D \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

The normalization $Z(\underline{\tau}_D)$ counts the number of solutions compatible with the partial assignment of variables in D.

A possible procedure for sampling from $\mu(\cdot)$ goes as follows. Choose arbitrarily a permutation of $\{1, \ldots, N\}$, denoted $i(1), \ldots, i(N)$, and call $D_t = \{i(1), \ldots, i(t)\}$ for $t = 1, \ldots, N$, $D_0 = \emptyset$. Construct now sequentially a configuration $\underline{\tau}$, assigning at time t the value $\tau_{i(t)}$; to this aim draw $\sigma_{i(t)}$ according to the marginal of the conditioned measure $\mu(\cdot|\underline{\tau}_{D_{t-1}})$, and set $\tau_{i(t)} = \sigma_{i(t)}$. It is easy to see that after t steps of the algorithm the partial configuration $\underline{\tau}_{D_t}$ is distributed according to the marginal law of $\mu(\cdot)$. In particular the final configuration $\underline{\tau}$ obtained when the N variables are assigned is a uniformly chosen solution of the CSP.

This simple algorithm would thus provide a uniform sampler of the solution set of any CSP; it is, however, only meant as a thought experiment. Indeed, computing exactly the probabilities $\mu(\sigma_{i(t)}|\underline{\tau}_{D_{t-1}})$ is in general a #P-complete problem, with no polynomial algorithm known until now, and we shall thus content ourselves with faster yet approximate means for computing these marginal probabilities. Before introducing them let us discuss further the idealized procedure.

J. Stat. Mech. (2009) P09001

The analytical description of the dynamics followed by this ideal process seems very difficult: at each time step the probability of the evolution $\underline{\tau}_{D_{t-1}} \to \underline{\tau}_{D_t}$ depends in a non-trivial way on all the choices made in the previous steps. However the description of the process at a given point of its evolution is very simple. As noted above $\underline{\tau}_{D_t}$ is distributed according to the marginal of $\mu(\cdot)$. One can state this in a slightly different way: $\underline{\tau}_{D_t}$ can be obtained by drawing uniformly a solution $\underline{\tau}$ from $\mu(\cdot)$, retaining the configuration of the variables in D_t, and discarding the rest of the configuration. We shall further assume that the permutation $i(1), \ldots, i(N)$ is drawn uniformly at random, such that D_t is a random set of t variables among N. In the thermodynamic limit we shall define $\theta = t/N$, the fraction of assigned variables, and consider for simplicity that $D_{\theta N}$ is built by retaining independently each variable with probability θ (we only make an error of order $1/\sqrt{N}$ on the fraction of variables thus included in D).

These considerations lead us to the definition of an ensemble of CSP instances parameterized by α and θ, generalizing the original one which corresponds to $\theta = 0$. Explicitly this ensemble of formulae corresponds to the following generation process:

(1) draw a satisfiable CSP with parameter α;
(2) draw a uniform solution $\underline{\tau}$ of this CSP;
(3) choose a set D by retaining each variable independently with probability θ;
(4) consider the residual formula on the variables outside D obtained by imposing the allowed configurations to coincide with $\underline{\tau}$ on D.

Let us emphasize that, apart from simple cases like the xorsat model, these ensembles do not coincide in general with randomly uniform formulae conditioned on their degree distributions. The fact that the generation of the configuration $\underline{\tau}$ depends on the initial CSP induces non-trivial correlations in the structure of the final formula.

We shall see in the following how to adapt the statistical mechanics techniques to compute the typical properties of such generalized formulae, and in particular to determine the phase transition thresholds in the (α, θ) plane. One characterization of these random ensembles is the quenched average residual entropy:

$$\omega(\theta) = \lim_{N\to\infty} \frac{1}{N} \mathbb{E}_F \mathbb{E}_{\underline{\tau}} \mathbb{E}_D [\ln Z(\underline{\tau}_D)], \qquad Z(\underline{\tau}_D) = \sum_{\underline{\sigma}} \prod_{a=1}^{M} \psi_a(\underline{\sigma}_{\partial a}) \mathbb{I}(\underline{\sigma}_D = \underline{\tau}_D), \tag{5}$$

where the three expectation values correspond to the three steps of the definition above. This quantity is similar, yet distinct, from the Franz–Parisi quenched potential [26]. The definition of the latter also involves a 'thermalized' reference configuration $\underline{\tau}$, but is given by the free energy of the measure on the configurations at a given Hamming distance from $\underline{\tau}$. In other words the two real replicas $\underline{\sigma}$ and $\underline{\tau}$ are coupled uniformly across the variables in a Franz–Parisi quenched potential, whereas in the definition of ω they are coupled infinitely strongly on D where they are forced to coincide, and not at all outside D. The computations presented in the rest of this paper can, however, be easily adapted to obtain the usual quenched potential.

We shall characterize the reduced measure $\mu(\cdot|\underline{\tau}_D)$ more precisely by computing other quantities besides $\omega(\theta)$. The existence of clusters in this measure will be tested by the computation of the long-range point-to-set correlations and the complexity of the typical clusters.

J. Stat. Mech. (2009) P09001

2.3. Bethe–Peierls approximation for decimated CSPs

We recall in this section the Bethe–Peierls approximation for statistical models defined on factor graphs and show how to adapt it to partially decimated CSPs. Let us first consider a probability measure with a weight function which can be factorized as in equation (3), with ψ_a some *a priori* arbitrary positive functions. The Bethe approximation for the computation of the partition function Z consists in extremizing the following expression:

$$\ln Z = -\sum_{i,a\in\partial i} \ln\left(\sum_{\sigma_i} \nu_{a\to i}(\sigma_i)\eta_{i\to a}(\sigma_i)\right) + \sum_a \ln\left(\sum_{\underline{\sigma}_{\partial a}} \psi_a(\underline{\sigma}_{\partial a}) \prod_{i\in\partial a} \eta_{i\to a}(\sigma_i)\right) + \sum_i \ln\left(\sum_{\sigma_i} \prod_{a\in\partial i} \nu_{a\to i}(\sigma_i)\right) \tag{6}$$

over the unknown $\{\nu_{a\to i}, \eta_{i\to a}\}$. These are probability measures on the alphabet of σ_i, defined on the directed edges of the factor graph, which we shall call messages for reasons that will become clear below. The extremization of the Bethe approximation for $\ln Z$ leads to a set of equations between the messages:

$$\nu_{a\to i}(\sigma_i) = f(\{\eta_{j\to a}\}_{j\in\partial a\setminus i}), \qquad \eta_{i\to a}(\sigma_i) = g(\{\nu_{b\to i}\}_{b\in\partial i\setminus a}), \tag{7}$$

where the (edge-dependent) functions f and g are defined by

$$\nu_{a\to i}(\sigma_i) = \frac{1}{z_{a\to i}} \sum_{\underline{\sigma}_{\partial a\setminus i}} \psi_a(\underline{\sigma}_{\partial a}) \prod_{j\in\partial a\setminus i} \eta_{j\to a}(\sigma_j), \qquad \eta_{i\to a}(\sigma_i) = \frac{1}{z_{i\to a}} \prod_{b\in\partial i\setminus a} \nu_{b\to i}(\sigma_i), \tag{8}$$

with $z_{a\to i}$ and $z_{i\to a}$ ensuring the normalization of $\nu_{a\to i}$ and $\eta_{i\to a}$. When the factor graph is a tree the log partition function is exactly given by (6) evaluated on the unique solution of the stationarity equations (8), see for instance [24]. The messages $\nu_{a\to i}$ (resp. $\eta_{i\to a}$) are then the marginal probabilities for σ_i of a modified measure corresponding to a factor graph where all factor nodes around i except a have been removed (resp. only a has been removed). From the knowledge of the messages solution of (8) one can compute the marginal probability of the variables in the full factor graph law (3), for instance the marginal probability of variable i is

$$\frac{1}{z_i} \prod_{a\in\partial i} \nu_{a\to i}(\sigma_i), \tag{9}$$

with again z_i fixed by normalization. In general factor graphs do contain loops, in that case (6), (8), (9) are only approximations, at the basis of the so-called belief propagation algorithm discussed in more details below.

The Bethe approximation can be easily adapted to the case where the configuration is forced to the value $\underline{\tau}_D$ on a subset of the sites $i \in D$, that is to the conditional measure (4). The estimation of the conditioned log partition function follows from (6)

$$\ln Z(\underline{\tau}_D) = -\sum_{i\notin D,a\in\partial i} \ln\left(\sum_{\sigma_i} \nu^{\underline{\tau}_D}_{a\to i}(\sigma_i)\eta^{\underline{\tau}_D}_{i\to a}(\sigma_i)\right) + \sum_a \ln\left(\sum_{\underline{\sigma}_{\partial a}} \psi_a(\underline{\sigma}_{\partial a}) \prod_{i\in\partial a} \eta^{\underline{\tau}_D}_{i\to a}(\sigma_i)\right) + \sum_{i\notin D} \ln\left(\sum_{\sigma_i} \prod_{a\in\partial i} \nu^{\underline{\tau}_D}_{a\to i}(\sigma_i)\right), \tag{10}$$

J. Stat. Mech. (2009) P09001

where the messages $\{\eta_{i\to a}^{\underline{\tau}_D}, \nu_{a\to i}^{\underline{\tau}_D}\}$ depend on the imposed partial configuration $\underline{\tau}_D$. They indeed obey the same equations (8), complemented with the boundary conditions $\eta_{i\to a}^{\underline{\tau}_D}(\sigma_i) = \delta_{\sigma_i,\tau_i}$ when $i \in D$.

2.4. Practical approximate implementation of the thought experiment

The ideal sampling algorithm described in section 2.2 cannot be practically implemented, because the computation of the marginals of the probability law $\mu(\underline{\sigma}|\underline{\tau}_D)$ has generically a cost exponential in the number of variables. One can, however, mimic this procedure, using a faster yet approximate estimation of the marginals of $\mu(\underline{\sigma}|\underline{\tau}_D)$ by means of the belief propagation algorithm. This modification of the ideal sampler, which will be called BP guided decimation in the following, thus corresponds to (for a given CSP instance):

(1) choose a random order of the variables, $i(1), \ldots, i(N)$, call $D_0 = \emptyset$, $D_t = \{i(1), \ldots, i(t)\}$;
(2) for $t = 1, \ldots, N$:
   (a) find a fixed point of the BP equations (7) with the boundary conditions $\eta_{i\to a}(\sigma_i) = \delta_{\sigma_i,\tau_i}$ when $i \in D_{t-1}$;
   (b) draw $\sigma_{i(t)}$ according to the BP estimation of $\mu(\sigma_i|\underline{\tau}_{D_{t-1}})$ given in (9);
   (c) set $\tau_{i(t)} = \sigma_{i(t)}$.

The belief propagation part of the algorithm corresponds to step 2(a). It amounts to searching for a stationary point of the Bethe approximation for the log partition function, in an iterative manner. The unknowns of the Bethe expression, $\eta_{i\to a}$ (resp. $\nu_{a\to i}$), are considered as messages passed from a variable to a neighboring clause (resp. from a clause to a variable). In a random sequential order a message, say $\eta_{i\to a}$, updates itself by recomputing its value from the current messages sent by its neighbors $\{\nu_{b\to i}\}_{b\in\partial i\setminus a}$, according to the equation in (8). If the factor graph of the formula was a tree, these iterations would converge in a finite number of updates to the unique fixed point solution of (8). On generic factor graphs there is no guarantee of convergence of these iterations, in practical implementations one has thus to precise the definition of the algorithm, giving criteria to stop the iterations of the BP updates; we shall come back to this point in section 5.5.

The definition of the probability measure conditioned on the choice of the reference configuration $\underline{\tau}_D$ (4) and the subsequent derivation of the BP equations only make sense if the formula admits at least one solution compatible with $\underline{\tau}_D$. In the analysis of the ideal algorithm this is automatically the case as soon as the initial formula is satisfiable. However, this can fail in the course of the BP guided decimation algorithm because the marginals used to generate the configuration $\underline{\tau}$ are only approximate. The BP equations are no longer well defined when there are no solutions of the formula compatible with $\underline{\tau}_D$. This shows up, for instance, in the computation of the message sent by a variable i to a clause a; whenever the product $\prod_{b\in\partial i\setminus a} \nu_{b\to i}(\sigma_i)$ vanishes for all possible values of σ_i the message $\eta_{i\to a}$ can no longer be normalized; a contradiction has occurred between the strong requests imposed by the clauses in $\partial i \setminus a$. The BP guided decimation algorithm has then to stop and fails to construct a solution of the formula.

This mechanism which unveils the contradictions in the choice of $\underline{\tau}_D$, and the fact that no solution is compatible with it, is actually equivalent to the unit clause propagation

J. Stat. Mech. (2009) P09001

(UCP) algorithm well known in computer science. For concreteness let us recall its functioning in the case of satisfiability formulae. UCP takes in input a list of variables and a list of clauses. If all clauses have length greater or equal to two it stops. Otherwise it chooses one of the unit clauses (i.e. of length 1). The variable in this unit clause must be fixed to the value satisfying the clause for the constructed configuration of variables to be a solution of the formula. The logical implications of this assignment are then drawn. All the clauses where the fixed variable appeared with the same sign as in the unit clause can be removed from the formula, as they are automatically satisfied. All clauses where it appeared with the opposite sign are effectively reduced in length, as the fixed variable will never satisfy them. This process is iterated as long as unit clauses are present in the formula. If clauses of length 0 are produced during the propagation of logical implications, then the input formula was not satisfiable: the logical implications required at least one of the variables to take simultaneously its two possible values. If no contradictions occur, the set of variables that appeared in unit clauses during the process are termed logically implied, their value being uniquely determined by the input formula.

It turns out that an equivalent formulation of the UCP rule for drawing logical implications can be given in terms of a message-passing procedure known as warning propagation (WP) [16]. The messages $\{\mathfrak{n}_{i\to a}, \mathfrak{v}_{a\to i}\}$ that WP sends along edges of the factor graph are projections of those of BP, where the only information retained is whether a value of the variable σ_i is authorized ($\eta_{i\to a}(\sigma_i) > 0$) or not ($\eta_{i\to a}(\sigma_i) = 0$):

$$\mathfrak{n}_{i\to a}(\sigma_i) = \mathbb{I}(\eta_{i\to a}(\sigma_i) > 0), \qquad \mathfrak{v}_{a\to i}(\sigma_i) = \mathbb{I}(\nu_{a\to i}(\sigma_i) > 0). \tag{11}$$

The projection of the BP equations (7) leads to recurrence equations on the WP messages

$$\mathfrak{v}_{a\to i} = \mathfrak{f}(\{\mathfrak{n}_{j\to a}\}_{j\in\partial a\setminus i}), \qquad \mathfrak{n}_{i\to a} = \mathfrak{g}(\{\mathfrak{v}_{b\to i}\}_{b\in\partial i\setminus a}). \tag{12}$$

Monotonicity arguments can be used to show that these recurrence equations, initialized with the permissive value of the messages $\mathfrak{n}_{i\to a}(\sigma_i) = 1$, converge to a unique fixed point independent on the order of updates of the messages. Moreover this fixed point contains the same information as revealed by UCP: a contradiction occurs in UCP if and only if there is a variable i such that

$$\prod_{a\in\partial i} \mathfrak{v}_{a\to i}(\sigma_i) = 0 \qquad \forall\sigma_i. \tag{13}$$

If no contradiction occurs the set of variables logically implied by UCP corresponds to the variables i such that there is only one authorized value σ_i for it,

$$\exists!\,\sigma_i\colon \prod_{a\in\partial i} \mathfrak{v}_{a\to i}(\sigma_i) = 1. \tag{14}$$

This correspondence was explicitly shown in the case of satisfiability formulae in [22]. In our practical implementation of the BP guided decimation algorithm we check, after each assignment of a variable $\tau_i(t)$, whether a contradiction in the partial configuration $\underline{\tau}_{D_t}$ can be detected by UCP/WP. According to the pseudocode above we do not immediately assign the variables which are logically implied by $\underline{\tau}_{D_t}$; please note, however, that this does not modify at all the subsequent steps of the algorithm, as the BP equations effectively take into account the effect of these logical implications. We also emphasize the fact that in general there are variables which can only take one value under the law $\mu(\cdot|\underline{\tau}_D)$, yet

J. Stat. Mech. (2009) P09001

that are not unveiled as logically implied by UCP/WP; if this were the case UCP would always be successful on any satisfiable formula (and incidentally one would have P = NP), which is of course well known to be wrong.

As a final remark on the BP guided decimation algorithm, let us emphasize another difference with the theoretical analysis. In practice we apply the algorithm to a uniformly generated formula of CSP, with $\alpha < \alpha_{\rm s}$, which are typically satisfiable in the thermodynamic limit, but we cannot systematically exclude unsatisfiable instances as in the analysis of the ideal algorithm.

3. Analysis of the thought experiment with the cavity method

3.1. Reminder on the usual cavity method

The goal of the cavity method is to compute the typical properties in the thermodynamic limit of graphical models defined on random factor graphs, and in particular the average entropy of the associated random CSP. In the simplest situation, known as the replica symmetric (RS) case, the hypothesis of the method is that the Bethe–Peierls approximation is asymptotically exact for the large random factor graphs of the ensemble considered. For concreteness we explain it in a setting encompassing the k-(xor)sat formulae, that is where each of the $M = \alpha N$ constraints imply k randomly chosen variables. The prediction for the average log partition function then follows by averaging (6):

$$\begin{aligned}\lim_{N\to\infty}\frac{1}{N}\mathbb{E}[\ln Z] = &-\alpha k\mathbb{E}\left[\ln\left(\sum_{\sigma}\nu(\sigma)\eta(\sigma)\right)\right]\\ &+\alpha\mathbb{E}\left[\ln\left(\sum_{\sigma_1,\ldots,\sigma_k}\psi(\sigma_1,\ldots,\sigma_k)\eta_1(\sigma_1)\cdots\eta_k(\sigma_k)\right)\right]\\ &+\mathbb{E}\left[\ln\left(\sum_{\sigma}\nu_1(\sigma)\cdots\nu_l(\sigma)\right)\right].\end{aligned}\tag{15}$$

Let us detail the justification and the meaning of the right-hand side of this relation. We have first used the translational invariance in the definition of the random ensemble: each term in the sums of equation (6) contributes on average in the same way. Moreover we have introduced the random variable η (resp. ν), whose distribution can be constructed as follows: drawing a random factor graph, finding the solution of the BP equations (7), picking a random edge i–a of the factor graph, and setting $\eta = \eta_{i\to a}$ (resp. $\nu = \nu_{a\to i}$). The averages on the right-hand side of (15) are thus over independent copies of η and ν, over the random constraint function ψ, and over a Poisson random variable l of mean αk. This last quantity is the degree of an uniformly chosen variable inside such a factor graph.

The equations fixing the distributions of the random variables η and ν can be obtained either looking for the stationary points of (15) or interpreting the BP equations (7) in the random graph perspective. Both reasoning leads to the distributional equations

$$\nu \stackrel{\rm d}{=} f(\eta_1,\ldots,\eta_{k-1}),\qquad \eta \stackrel{\rm d}{=} g(\nu_1,\ldots,\nu_l).\tag{16}$$

These equations have to be interpreted as equalities between distributions of random variables. The η_i (resp. ν_i) are independent copies of the random variable η (resp. ν), l is a Poissonian random variable of parameter αk, and f and g have been defined in (7) and (8), f being itself random because of the choice of the constraint function ψ.

The RS cavity method is based on the assumption of asymptotic correctness of the Bethe–Peierls approximation, which can be rephrased as the existence of a single pure state (or cluster) in the probability law μ. A complementary statement of the RS method concerns the local description of the law μ. Let us consider an arbitrary variable node i_0, and its depth L neighborhood, that is the set of variables at graph distance smaller than or equal to L from i_0. In such random graph ensembles this neighborhood is, with high probability, a Galton–Watson random tree with Poissonian offspring of mean αk for the variable nodes (the constraint nodes being of course always of degree k). In the RS case the marginal of the law μ for the variables in this neighborhood converges in distribution to the law of a finite tree of depth L, the only effect of the rest of the graph being summarized in a single message η acting on the boundary (the depth L variables), drawn independently with the fixed point distribution solution of equation (16). In other words the depth L variables would be considered independent if the constraints inside the depth L neighborhood around i_0 were removed.

In the context of random CSPs the RS assumption is only valid for small values of the density of constraints α. When this parameter grows the solution space splits into a large number of pure states (clusters), which induces non-trivial correlations between the variables of the graphical model. The cavity method at the level of the first step of replica symmetry breaking (1RSB) is able to describe this situation [32]. Let us sketch some of its important features without entering into technical details. The assumption of the 1RSB method is that the Bethe–Peierls approximation can still be used, but only for the probability measures restricted to a single pure state; to each such pure state is associated a solution of the BP equations (7). In order to handle the proliferation of pure states one introduces, on each directed edge of a given factor graph, a probability distribution, with respect to the choice of the pure states, of the corresponding BP messages. The 1RSB equations linking these distributions on adjacent edges depend on the Parisi parameter m, which controls the relative importance of the pure states in the sampling of the corresponding BP messages, according to their internal sizes. A slightly more explicit interpretation of the 1RSB equations has been given in [11, 30]. There it was shown that the distribution over pure states can be viewed as a distribution over 'far-away' boundary conditions, with a specific probability measure over these boundary conditions depending on m. A special role is played by the value $m = 1$. In this case the 'far-away' boundary conditions are themselves drawn from the original Gibbs measure. The clustering transition, that is the appearance of a non-trivial solution of the 1RSB equations at $m = 1$, can thus be related to the existence of long-range correlations of a particular type (so-called point-to-set) in the Gibbs measure, as first discussed in [31]. These correlations measure the influence on a variable i_0 of fixing a subset B of variables, for instance $B(i_0, \ell)$ the set of variables at distance exactly ℓ from i_0, to a reference value drawn from the equilibrium probability measure. Using the notation of conditional probability defined in equation (4) the typical long-range point-to-set correlation is

$$C_\infty = \lim_{\ell\to\infty} \lim_{N\to\infty} \mathbb{E} \sum_{\sigma_{i_0}} |\mu(\sigma_{i_0}|\underline{\tau}_{B(i_0,\ell)}) - \mu(\sigma_{i_0})|, \tag{17}$$

J. Stat. Mech. (2009) P09001

where the expectation is over the equilibrium configuration $\underline{\tau}$ and the factor graph model. In an unclustered regime the influence of the distant boundary vanishes, $C_\infty = 0$, while in the presence of clustering $C_\infty > 0$. In the latter case one can moreover compute the complexity of relevant clusters, that is the logarithm of the degeneracy of the clusters which bears the vast majority of the weights of the probability measures (at the leading exponential precision) by computing the difference in entropy of the conditional and original measures, $\mu(\cdot|\underline{\tau}_{B(i_0,\ell)})$ and $\mu(\cdot)$. The so-called condensation transition is signaled by a vanishing of this complexity. We shall come back in the next subsection to the technical details of the 1RSB $m = 1$ computations, generalizing it to the case of partially decimated formulae.

3.2. Cavity method for decimated CSP

We turn now to the extension of the cavity method to partially decimated factor graphs. Our goal is to compute the residual entropy (5) by averaging the Bethe expression (10) with respect to the distribution of the conditional messages $\{\eta_{i\to a}^{\underline{\tau}_D}, \nu_{a\to i}^{\underline{\tau}_D}\}$. The randomness of these objects has several origins: (i) the choice of the factor graph, (ii) the generation of the reference configuration $\underline{\tau}$ from the uniform measure μ and (iii) the selection of the decimated variables in D, each independently with probability θ. The difficulty of this computation arises from the correlation between (i) and (ii), the measure μ being itself defined in terms of the factor graph. This dependence can, however, be handled within the context of the cavity method. Let us suppose indeed that the local properties of the original measure $\mu(\cdot)$ are well described by the assumptions of replica symmetry, that is for $\alpha < \alpha_c$.[3] We can thus perform the computation in the random tree model that corresponds locally to the random graph one. In this case the generation of the reference configuration $\underline{\tau}$ of a tree factor graph model can be done recursively, in a broadcasting way, thanks to the Markov property of such probability laws. The value of the root τ_{i_0} is first drawn with its marginal probability computed from the incoming messages according to (9). Then the configuration of the neighbors of i_0 can be drawn conditioned on the value of τ_{i_0}, and the process can be iterated away from the root. Once the reference configuration $\underline{\tau}$ has been generated in such a way, the messages $\{\eta_{i\to a}^{\underline{\tau}_D}, \nu_{a\to i}^{\underline{\tau}_D}\}$ can be computed, their dependence on $\underline{\tau}$ arising from the condition $\eta_{i\to a}^{\underline{\tau}_D}(\sigma_i) = \delta_{\sigma_i,\tau_i}$ for variables i in the decimated set D. At this point, for a given tree, reference configuration $\underline{\tau}$ and set D, each directed edge of the factor graph bears a pair of messages, for instance $(\nu_{a\to i}, \nu_{a\to i}^{\underline{\tau}_D})$ on the edge from constraint a to variable i. We can now define the random variable $(\nu, \nu^\tau)_\ell$ which has the distribution of $(\nu_{a\to i}, \nu_{a\to i}^{\underline{\tau}_D})$ when one takes into account the randomness in the choice of the tree, of the set D and of the reference configuration $\underline{\tau}$, the latter being conditioned on $\tau_i = \tau$. Moreover the positive integer ℓ indexes the depth of the random tree construction. We shall also introduce random variables having the same distribution as $(\eta_{i\to a}, \eta_{i\to a}^{\underline{\tau}_D})$. For the sake of clarity in the following we actually introduce two versions of these random variables, $(\eta, \eta^\tau)_\ell$ and $(\eta, \widetilde{\eta}^\tau)_\ell$, the former being additionally conditioned on $i \notin D$. The equations defining these random variables by recurrence on ℓ can now easily be written:

$$(\eta, \widetilde{\eta}^\tau)_\ell \stackrel{\mathrm{d}}{=} \begin{cases} (\eta, \eta^\tau)_\ell & \text{with probability } 1-\theta \\ (\eta, \delta^\tau) & \text{otherwise,} \end{cases} \tag{18}$$

[3] The local properties are indeed of a RS type also in the clustered uncondensed regime $\alpha \in [\alpha_d, \alpha_c]$ [11].

J. Stat. Mech. (2009) P09001

where we defined $\delta^\tau(\sigma) = \delta_{\tau,\sigma}$ and

$$(\eta, \eta^\tau)_\ell \stackrel{\mathrm{d}}{=} (g(\nu_1, \ldots, \nu_l), g(\nu_1^\tau, \ldots, \nu_l^\tau)), \tag{19}$$

where l is a Poisson random variable of parameter αk and $(\nu_1, \nu_1^\tau), \ldots, (\nu_l, \nu_l^\tau)$ are independent copies of $(\nu, \nu^\tau)_\ell$. Finally one has

$$(\nu, \nu^\tau)_{\ell+1} \stackrel{\mathrm{d}}{=} (f(\eta_1, \ldots, \eta_{k-1}), f(\widetilde{\eta}_1^{\tau_1}, \ldots, \widetilde{\eta}_{k-1}^{\tau_{k-1}})), \tag{20}$$

where the $(\eta_i, \widetilde{\eta}_i^{\tau_i})$ are independent copies of $(\eta, \widetilde{\eta}^{\tau_i})_\ell$, and the configuration $\tau_1, \ldots, \tau_{k-1}$ is drawn according to

$$P[\tau_1, \ldots, \tau_{k-1}|\tau] = \frac{1}{z}\psi(\tau, \tau_1, \ldots, \tau_{k-1})\eta_1(\tau_1)\cdots\eta_{k-1}(\tau_{k-1}). \tag{21}$$

Let us emphasize that the function ψ used in the broadcasting generation of $\tau_1, \ldots, \tau_{k-1}$ is the same (random) constraint function as the one used to compute f in (20), and that the messages η_i are the same in (20) and (21). We shall discuss a numerical procedure for solving these equations on the example of satisfiability formulae in section 5. The prediction of the cavity method for the average residual entropy (5) can finally be expressed in terms of these random variables:

$$\begin{aligned}\omega = &-\alpha k(1-\theta)\mathbb{E}\left[\sum_\tau \frac{\nu(\tau)\eta(\tau)}{\sum_{\tau'}\nu(\tau')\eta(\tau')}\ln\left(\sum_\sigma \nu^\tau(\sigma)\eta^\tau(\sigma)\right)\right] \\ &+ \alpha\mathbb{E}\left[\sum_{\tau_1,\ldots,\tau_k}\frac{\psi(\tau_1,\ldots,\tau_k)\eta_1(\tau_1)\cdots\eta_k(\tau_k)}{\sum_{\tau_1',\ldots,\tau_k'}\psi(\tau_1',\ldots,\tau_k')\eta_1(\tau_1')\cdots\eta_k(\tau_k')}\right. \\ &\left.\times\ \ln\left(\sum_{\sigma_1,\ldots,\sigma_k}\psi(\sigma_1,\ldots,\sigma_k)\widetilde{\eta}_1^{\tau_1}(\sigma_1)\cdots\widetilde{\eta}_k^{\tau_k}(\sigma_k)\right)\right] \\ &+ (1-\theta)\mathbb{E}\left[\sum_\tau\frac{\nu_1(\tau)\cdots\nu_l(\tau)}{\sum_{\tau'}\nu_1(\tau')\cdots\nu_l(\tau')}\ln\left(\sum_\sigma\nu_1^\tau(\sigma)\cdots\nu_l^\tau(\sigma)\right)\right],\end{aligned} \tag{22}$$

where as before l is a Poisson random variable of parameter αk and the various $(\eta_i, \eta_i^{\tau_i})$ are independent copies of the corresponding random variables, with the $\ell \to \infty$ limit kept understood.

We shall now discuss an important issue that we kept under silence, namely the definition of the initial condition $(\eta, \eta^\tau)_{\ell=0}$. Let us first make the connection between the computation just presented and the $m = 1$ 1RSB description of non-decimated formulae, that is consider for a while the case $\theta = 0$. We shall call I_0 the initial condition $(\eta, \eta^\tau)_{\ell=0} \stackrel{\mathrm{d}}{=} (\eta, \eta)$, with η a random variable solution of the RS fixed point equation (16). It is easy to show that with such an initial condition $(\eta, \eta^\tau)_\ell \stackrel{\mathrm{d}}{=} (\eta, \eta)$ for all values of ℓ, and that (22) reduces to the RS prediction (15) for the average entropy. If on the contrary one considers the initial condition I_1 defined by $(\eta, \eta^\tau)_{\ell=0} \stackrel{\mathrm{d}}{=} (\eta, \delta^\tau)$ and iterates the recursion equations (18)–(20), one reproduces the $m = 1$ computations in the form of [30, 31]. This should not be a surprise: the interpretation of the $m = 1$ 1RSB formalism presented above was precisely the study of the effect of a far-away boundary, which is implemented here in this initial condition and in the limit $\ell \to \infty$. In an unclustered phase the two initial conditions lead to the same random variable (η, η) in the large ℓ limit, the effect

J. Stat. Mech. (2009) P09001

of the far-away boundary vanishes at long distance and the correlation function (17) is equal to 0. If, in contrast, the two initial conditions do not lead to the same limit when ℓ diverges the correlation function remains positive, signaling the presence of clustering in the solution space. In that case the value of (22) computed with the initial condition I_1 is the internal entropy of the relevant clusters, the difference between (15) and (22) is ascribed to the complexity, the degeneracy of the relevant clusters. We now come back to the decimated case $\theta > 0$. The two initial conditions are still relevant and have the same interpretation with the measure $\mu(\cdot)$ replaced by $\mu(\cdot|\underline{\tau}_D)$. Suppose indeed that the point-to-set correlation criterion were to be tested for the typical properties of $\mu(\cdot|\underline{\tau}_D)$. Then one would compute the generalization of (17):

$$C_\infty(\theta) = \lim_{\ell\to\infty}\lim_{N\to\infty}\mathbb{E}\sum_{\sigma_{i_0}}|\mu(\sigma_{i_0}|\underline{\tau}_{D\cup B(i_0,\ell)}) - \mu(\sigma_{i_0}|\underline{\tau}_D)|, \tag{23}$$

where the expectation would include an average over the set D of θN variables. This is precisely what is realized by the two initial conditions followed by the recursion relations (18)–(20) for this value of θ. Again we shall conclude on the absence of clustering in $\mu(\cdot|\underline{\tau}_D)$ when the two initial conditions have the same $\ell\to\infty$ limit, and obtain a typical complexity of the clusters from the difference in the values of (22) for the two different initial conditions.

3.3. The cavity computation of the average number of logically implied variables

Another quantity which can be interesting to compute is the amount of logical implications, in the UCP/WP sense explained in section 2.4, induced by the choice of the partial reference solution $\underline{\tau}_D$. Let us define as

$$\phi(\theta) = \theta + \lim_{N\to\infty}\frac{1}{N}\mathbb{E}_F\mathbb{E}_{\underline{\tau}}\mathbb{E}_D[\text{nb. directly implied variables}], \tag{24}$$

the average fraction of variables which have been explicitly assigned or which can be logically deduced from these assignments. A motivation for the study of this quantity, as explained in [22], is that $(\mathrm{d}\phi/\mathrm{d}\theta) - 1$ is the average number of newly implied variables as a single assignment step is performed. The size of this set of variables, and in particular its possible divergence with the formula size when $\phi(\theta)$ is discontinuous, should thus be a measure of the sensitivity of the decimation procedure with respect to small errors made by the BP version of the procedure with respect to the ideal one.

The computation of $\phi(\theta)$ can be performed, within the RS assumptions on the local structure of the probability law $\mu(\cdot)$, in the locally equivalent random tree model. For a generic CSP one can reproduce the reasoning above, replacing the conditional messages $\eta_{i\to a}^{\underline{\tau}_D}$ by their WP counterparts $\mathfrak{n}_{i\to a}^{\underline{\tau}_D}$ according to the projection defined in (11). This leads to the similar definition of sequences of random pairs of variables, $(\eta,\widetilde{\mathfrak{n}}^\tau)_\ell$, $(\eta,\mathfrak{n}^\tau)_\ell$ and $(\eta,\mathfrak{v}^\tau)_\ell$, which obeys recursion equations analogous to equations (18)–(20):

$$(\eta,\widetilde{\mathfrak{n}}^\tau)_\ell \overset{\mathrm{d}}{=} \begin{cases}(\eta,\mathfrak{n}^\tau)_\ell & \text{with probability } 1-\theta\\ (\eta,\delta^\tau) & \text{otherwise,}\end{cases}$$

$$(\eta,\mathfrak{n}^\tau)_\ell \overset{\mathrm{d}}{=} (g(\nu_1,\dots,\nu_l),\mathfrak{g}(\mathfrak{v}_1^\tau,\dots,\mathfrak{v}_l^\tau)), \tag{25}$$

$$(\nu,\mathfrak{v}^\tau)_{\ell+1} \overset{\mathrm{d}}{=} (f(\eta_1,\dots,\eta_{k-1}),\mathfrak{f}(\widetilde{\mathfrak{n}}_1^{\tau_1},\dots,\widetilde{\mathfrak{n}}_{k-1}^{\tau_{k-1}})). \tag{26}$$

J. Stat. Mech. (2009) P09001

The function $\phi(\theta)$ can then be computed as

$$\phi(\theta) = \mathbb{E}\left[\sum_{\tau} \eta(\tau)\mathbb{I}(\widetilde{\mathfrak{n}}^{\tau} = \delta^{\tau})\right], \tag{27}$$

in the $\ell \to \infty$ limit. This form of the computation is valid for any CSP; in some particular cases one can, however. devise a more efficient (and numerically more precise) formulation, using the specific form of the constraint rules to replace the WP message in the second entry of the random pair by a probability of implication. We refer the reader to [22] for further details on this computation for random satisfiability formulae.

4. Application to the XORSAT ensemble

We apply in this section the general formalism developed above to the ensemble of xor-satisfiability formulae. As we shall see great simplifications occur for this simple model, and the study of both the decimated ensemble of random formulae and of the BP guided decimation algorithm can in fact be performed with simpler methods [28, 29, 33]. It is, however, an instructive toy model to begin with before handling the more involved case of satisfiability formulae, and has deep connections with information theory, in particular with the low density parity check codes. The analysis of the 'Maxwell decoder' in [34] is actually very tightly related to the content of this section.

4.1. BP equations

The constraints of an xorsat CSP have been defined in equation (1). The set of solutions of such a CSP exhibits some simplifying symmetries, even in the decimated case where some variables are fixed to a given value. It is indeed easy to realize that a non-decimated variable is either fixed to $+1$ in all the solutions, or fixed to -1, otherwise it takes the value $+1$ in exactly half of the solutions and -1 in the other half. Consequently the BP messages $\nu_{a\to i}$ and $\eta_{i\to a}$ can be restricted to a set of three possible types, which we shall encode with three-valued numbers $u_{a\to i}$ and $h_{i\to a}$ according to the following correspondence:

$$\nu_{a\to i}(\sigma_i) = \begin{cases} \delta_{\sigma_i,+1} & \Leftrightarrow u_{a\to i} = 1 \\ \delta_{\sigma_i,-1} & \Leftrightarrow u_{a\to i} = -1 \\ \frac{1}{2} & \Leftrightarrow u_{a\to i} = 0, \end{cases} \qquad \eta_{i\to a}(\sigma_i) = \begin{cases} \delta_{\sigma_i,+1} & \Leftrightarrow h_{i\to a} = 1 \\ \delta_{\sigma_i,-1} & \Leftrightarrow h_{i\to a} = -1 \\ \frac{1}{2} & \Leftrightarrow h_{i\to a} = 0. \end{cases} \tag{28}$$

With these notations the BP equations (8) can be rewritten as

$$u_{a\to i} = J_a \prod_{j\in\partial a\setminus i} h_{j\to a}, \qquad h_{i\to a} = \begin{cases} 0 & \text{if } u_{b\to i} = 0 \ \forall b \in \partial i \setminus a \\ +1 & \text{if } \exists b \in \partial i \setminus a \text{ with } u_{b\to i} = 1 \\ -1 & \text{if } \exists b \in \partial i \setminus a \text{ with } u_{b\to i} = -1. \end{cases} \tag{29}$$

The second expression is well defined as long as no contradictions are detected, which means that the conditions in the last two lines are not fulfilled simultaneously. The boundary condition for a decimated variable $i \in D$ is $h_{i\to a}^{\tau_D} = \tau_i$ for all neighboring interactions $a \in \partial i$.

Due to the symmetry of the model the BP equations (29) can actually be regarded as WP equations that express the simplifications of the unit propagation rule. The first

J. Stat. Mech. (2009) P09001

equation reflects the fact that a constraint imposes the value of one of its variables if and only if the $k-1$ other variables are fixed (either by decimation or by propagation of logical implications), while the second means that a variable is fixed as soon as one of its neighboring clauses imposes its value.

4.2. The cavity method computations

Following the general formalism introduced in section 3 for the treatment of decimated CSPs and the parameterization of the BP messages in terms of u and h, we have to find the distributions of the random variables $(h, h^\tau)_\ell$, $(u, u^\tau)_\ell$ and $(h, \widetilde{h}^\tau)_\ell$ for $\tau = \pm 1$. These are the solutions of equations (18)–(20) specialized to the functions f, g and ψ of the xorsat model. The relevant solution of this equation takes a particularly simple form which allows for an analytic solution. The results of [28, 29] imply indeed that the set of solutions of a non-decimated k-xorsat formula is described by the trivial solution of the RS equation, $u = h = 0$. Moreover the value of the conditional message h^τ cannot be equal to $-\tau$: by definition h^τ describes the measure where the reference solution has been drawn conditional on its value at the root being τ, whereas $h^\tau = -\tau$ would mean that all solutions compatible with the values of the reference on some subset of variables D have $-\tau$ at the root, which is in contradiction with the hypothesis. As a consequence the solution of (18)–(20) can be looked for under the form

$$\begin{aligned}
(h, h^\tau)_\ell &\stackrel{\mathrm{d}}{=} \begin{cases} (0,0) & \text{with probability } 1-x_\ell \\ (0,\tau) & \text{with probability } x_\ell, \end{cases} \\
(u, u^\tau)_\ell &\stackrel{\mathrm{d}}{=} \begin{cases} (0,0) & \text{w.p. } 1-y_\ell \\ (0,\tau) & \text{w.p. } y_\ell, \end{cases} \qquad (h, \widetilde{h}^\tau)_\ell \stackrel{\mathrm{d}}{=} \begin{cases} (0,0) & \text{w.p. } 1-\phi_\ell \\ (0,\tau) & \text{w.p. } \phi_\ell. \end{cases}
\end{aligned} \tag{30}$$

Inserting these forms in (18)–(20) leads to the following equations:

$$\phi_\ell = \theta + (1-\theta)x_\ell, \qquad x_\ell = 1 - \exp[-\alpha k y_\ell], \qquad y_{\ell+1} = \phi_\ell^{k-1}, \tag{31}$$

which can be closed under a single recursion equation on ϕ_ℓ:

$$\phi_{\ell+1} = \theta + (1-\theta)\left(1 - \mathrm{e}^{-\alpha k \phi_\ell^{k-1}}\right). \tag{32}$$

The fixed point equation $\phi_{\ell+1} = \phi_\ell$ has between one and three distinct solutions on $[0,1]$, depending on the values of α and θ (examples of the various situations are provided in figure 1). A quick analysis of the equation shows that for $\alpha < \alpha_*$, with

$$\alpha_* = \frac{1}{k}\left(\frac{k-1}{k-2}\right)^{k-2}, \tag{33}$$

equation (32) admits a single solution for all values of θ. If, in contrast, $\alpha > \alpha_*$, there exists a range of θ, denoted $[\theta_-(\alpha), \theta_+(\alpha)]$, where equation (32) admits three solutions in $[0,1]$. In that case we shall call $\phi(\theta)$ (resp. $\psi(\theta)$) the smallest (resp. the largest) of these three solutions. Some examples of these curves are shown in figure 2 and the lines $\theta_\pm(\alpha)$ are displayed in figure 3.

The expression of ω given in equation (22) can be computed using the ansatz (30)

$$\omega = (\ln 2)[\alpha k(1-\theta)(1-xy) - \alpha(1-\phi^k) - (1-\theta)((\alpha k)(1-y) - \exp[-\alpha k y])], \tag{34}$$

J. Stat. Mech. (2009) P09001

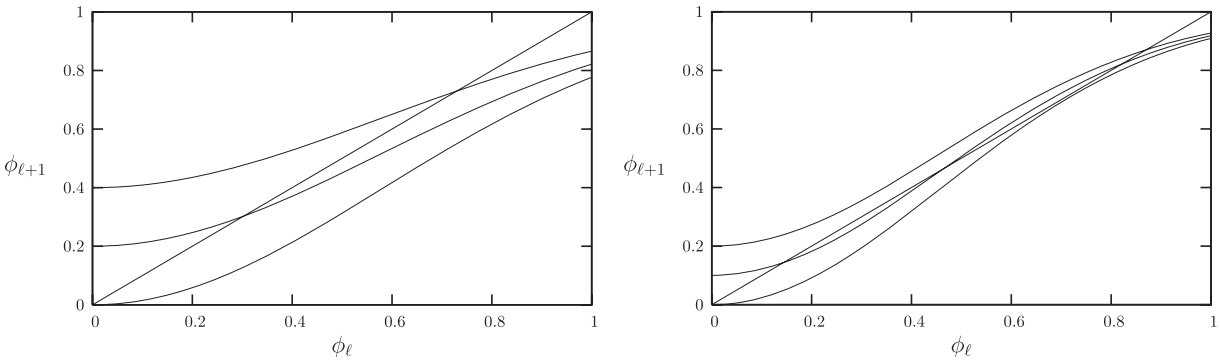

Figure 1. Illustration of the recurrence equation (32) for 3-xorsat. Left panel: $\alpha = 0.5 < \alpha_*(k = 3) = 2/3$, right panel: $\alpha = 0.8 > \alpha_*$. On each of the plots the three curves are for different values of θ (increasing from bottom to top).

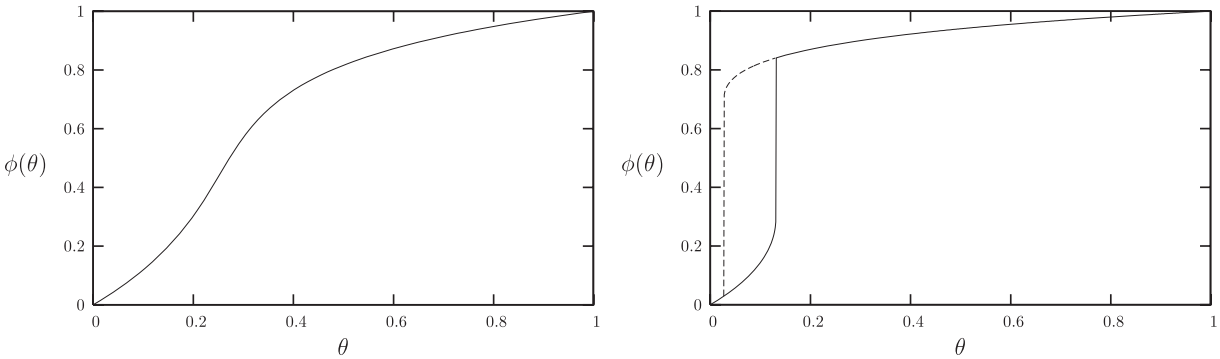

Figure 2. Fixed point solution(s) of equation (32) for $k = 3$. Left panel: $\alpha = 0.5 < \alpha_*(k = 3) = 2/3$, right panel: $\alpha = 0.8 > \alpha_*$. The solid line is $\phi(\theta)$, the dashed line in the right panel is $\psi(\theta)$ and the largest fixed point solution of equation (32).

where the limit $\ell \to \infty$ is kept as understood. Using the relations between x, y and ϕ stated in (31), one can express this residual entropy in terms of ϕ:

$$\widehat{\omega}(\phi) = (\ln 2)[1 - \phi - \alpha + \alpha k(1 - \phi)\phi^{k-1} + \alpha\phi^k]. \tag{35}$$

When $\alpha < \alpha_*$ the fixed point solution of (32) is unique, hence $\omega(\theta) = \widehat{\omega}(\phi(\theta))$ is a smoothly decreasing function of θ, as plotted on the left panel of figure 4. For larger values of α, i.e. $\alpha > \alpha_*$, we have seen above that there exists a range of parameters $\theta \in [\theta_-(\alpha), \theta_+(\alpha)]$ where two solutions of (32), $\phi(\theta)$ and $\psi(\theta)$, coexist. In the right panel of figure 4 one can see that the two branches of the entropy, $\widehat{\omega}(\phi(\theta))$ and $\widehat{\omega}(\psi(\theta))$, cross each other at an intermediate value $\theta_c(\alpha) \in [\theta_-(\alpha), \theta_+(\alpha)]$, which is also plotted as a function of α in the phase diagram figure 3. It is natural (and we shall argue in the following that it is the correct choice) to consider that in the region of coexistence the relevant branch is the one leading to the largest entropy, $\omega(\theta) = \max[\widehat{\omega}(\phi(\theta)), \widehat{\omega}(\psi(\theta))]$, which thus exhibits a discontinuity in its slope when θ crosses θ_c.

A direct justification of this choice will be given in the next subsection; here we argue in its favor on the basis of the cavity method. The two boundary conditions discussed

J. Stat. Mech. (2009) P09001

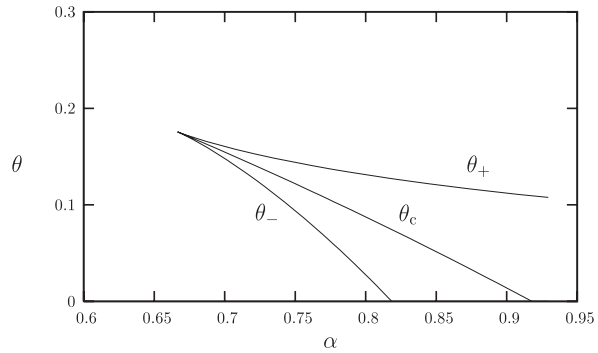

Figure 3. Phase diagram in the (α, θ) plane for the ensemble of 3-xorsat decimated random formulae; the three critical lines meet at $\alpha_*(k = 3) = 2/3$.

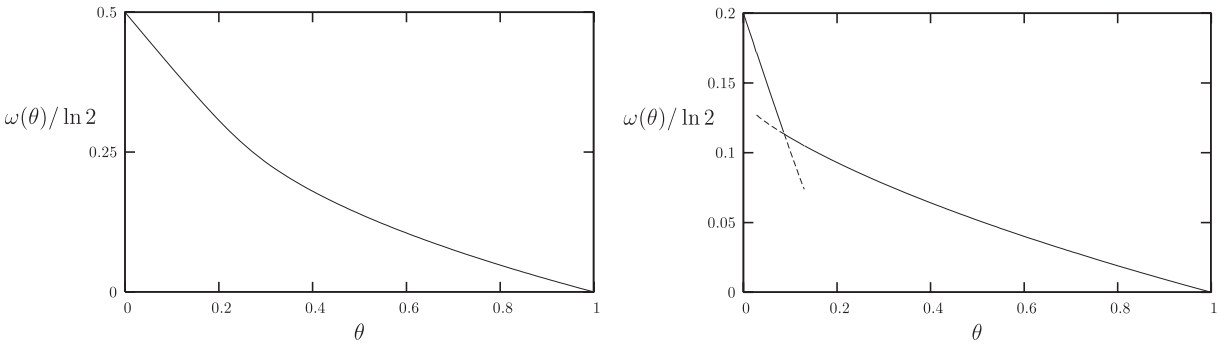

Figure 4. Residual entropy $\omega(\theta)$ for 3-xorsat. Left panel: $\alpha = 0.5$, right panel: $\alpha = 0.8$. The solid line is the true entropy $\omega(\theta) = \max[\widehat{\omega}(\phi(\theta)), \widehat{\omega}(\psi(\theta))]$, while the dashed lines are the two irrelevant branches, $\min[\widehat{\omega}(\phi(\theta)), \widehat{\omega}(\psi(\theta))]$.

in section 3.2 corresponds to $\phi_{\ell=0} = 0$ (I_0) and $\phi_{\ell=0} = 1$ (I_1). When the fixed point solution of (32) is unique both initial conditions lead to the same limit $\phi(\theta)$ in the large ℓ limit, which leads to the conclusion that there is no clustering in the solution space of the decimated formula for these values of α and θ.

In the region of coexistence these two initial conditions yield, respectively, $\phi_\ell \to \phi(\theta)$ and $\phi_\ell \to \psi(\theta)$ as ℓ diverges. One is thus led to assign the difference $\widehat{\omega}(\phi(\theta)) - \widehat{\omega}(\psi(\theta))$ to the complexity of the decimated formula, that is the contribution of the entropy due to the presence of clusters in the measure $\mu(\cdot|D_{\theta N})$. This interpretation is valid only when the complexity is positive, that is in the range $[\theta_-, \theta_c]$. A condensation transition occurs when the threshold θ_c is crossed. In the region $[\theta_c, \theta_+]$ only a subextensive number of clusters are relevant and the total entropy is equal to their internal entropy. The latter being given by the thermodynamic computation with the initial condition I_1, one concludes that $\omega(\theta) = \widehat{\omega}(\psi(\theta))$ for $\theta \in [\theta_c, \theta_+]$.

4.3. A more direct computation and its interpretation

It is instructive to rederive the above results on xor-satisfiability decimated formulae by more direct means. Let us first show that for this computation one can assume that the

J. Stat. Mech. (2009) P09001

formula is unfrustrated (i.e. $J_a = 1$ for all constraints) and that in the reference solution $\underline{\tau}$ all variables are fixed to $\tau_i = 1$. Suppose indeed that the formula has been generated with random $J_a = \pm 1$. As we have conditioned the ensemble of formulae on satisfiable ones, there is at least one solution, call it $\underline{\sigma}^{(0)}$. By the gauge transformation $\sigma_i \leftrightarrow \sigma'_i = \sigma_i \sigma_i^{(0)}$, the solutions $\underline{\sigma}$ of the original problem are in bijection with the solutions $\underline{\sigma}'$ of the unfrustrated model. Consider furthermore a reference solution $\underline{\tau}$ of the unfrustrated model and a set D of variables, such that the decimated problem to solve is

$$\prod_{i \in \partial a} \sigma'_i = 1 \qquad \forall a, \qquad \sigma'_i = \tau_i \qquad \forall i \in D. \tag{36}$$

Applying now the gauge transformation $\sigma'_i \leftrightarrow \sigma''_i = \sigma'_i \tau_i$, noting that $\underline{\tau}$ is a solution of the unfrustrated model, one reduces the problem to

$$\prod_{i \in \partial a} \sigma''_i = 1 \qquad \forall a, \qquad \sigma''_i = 1 \qquad \forall i \in D. \tag{37}$$

Having got rid of the signs in the constraints and in the reference solution, the size and the structure of the set of solutions of the decimated problem can be deduced from the underlying hypergraph of constraints [28, 29, 33].

The initial hypergraph is drawn uniformly with αN clauses of length k among N variables. A fraction θ of the variables are fixed to $+1$, and can thus be eliminated from the constraints which are reduced in size. Unit clause propagation can then be run to propagate these simplifications. The details of this computation are deferred to the appendix; we only quote here the results. When UCP stops, there are $N(1-\phi(\theta))$ variables unassigned, with $\phi(\theta)$ the smallest fixed point solution of (32). The simplified formula contains constraints of all lengths $\kappa \in [2, k]$; more precisely, there are $\alpha N \binom{k}{\kappa} (1-\phi)^\kappa \phi^{k-\kappa}$ clauses of length κ. The unassigned variables have a Poisson degree distribution with average $\alpha k (1 - \phi^{k-1})$.

At this point the structure of the solutions of this reduced formula can be studied with the leaf removal algorithm [28, 29]. The details are again deferred to the appendix. One finds that the presence of an extensive 2-core is equivalent to equation (32) admitting more than one solution, i.e. if $\alpha > \alpha_*$ and in the interval $\theta \in [\theta_-, \theta_+]$. If this is the case, the larger solution ψ gives the fraction of the variables which are either fixed at the end of UCP, or in the backbone of the UCP-reduced formula. The difference between the number of variables and the number of clauses in the 2-core is $N(\widehat{\omega}(\phi) - \widehat{\omega}(\psi))/(\ln 2)$. In the interval $\theta \in [\theta_-, \theta_{\rm c}]$ this quantity is positive, hence it is interpreted as the entropy of the number of solutions of the 2-core, i.e. the complexity of the reduced formula. In $[\theta_{\rm c}, \theta_+]$ the negative complexity is due to rare events. Typically the 2-core only contains a sub-exponential number of solutions, hence the discontinuity in slope of $\omega(\theta)$ at this condensation transition $\theta_{\rm c}$. For $\alpha \geq \alpha_{\rm d}$ (the usual dynamical threshold) the original formula already has a 2-core, hence $\theta_- = 0$ in this case. Similarly $\theta_{\rm c} = 0$ for $\alpha \geq \alpha_{\rm c}$, the satisfiability transition of the standard ensemble.

Let us remark that the density of clauses of length 2 in the UCP-reduced formula is $(1/2)\alpha k(k-1)(1-\phi)\phi^{k-2}$. When θ reaches θ_+ from below this density reaches $1/2$ and thus the sub-formula made of length 2 clauses percolates. Indeed θ_+ is the point of disappearance of the solution $\phi(\theta)$ from equation (32), hence by the implicit function theorem the derivatives with respect to ϕ of the two sides of equation (32) are equal at that point.

J. Stat. Mech. (2009) P09001

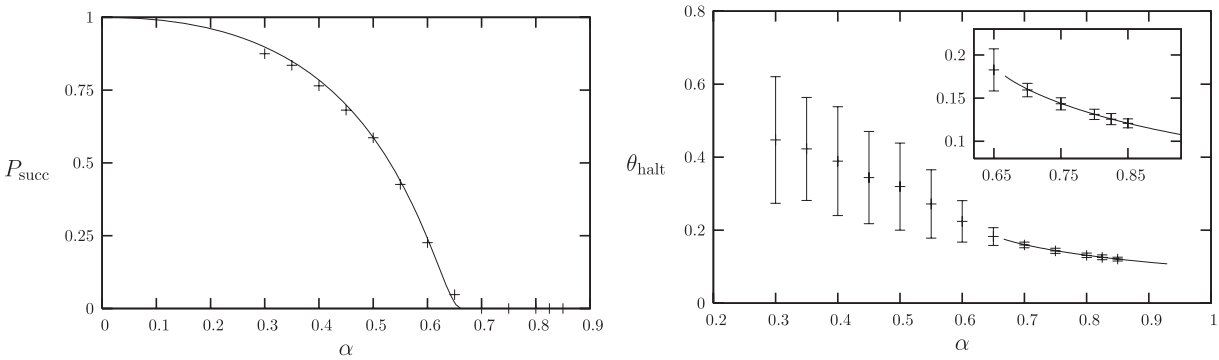

Figure 5. Left panel: probability of success of BP guided decimation on 3-xorsat random formulae. The solid line is the analytical prediction [36, 37] in the infinite size limit (cf equation (39)), which vanishes for $\alpha > \alpha_*$ and the symbols are the results of numerical simulations on 800 formulae of size $N = 2 \times 10^4$ variables for each value of α. Right panel: the solid line is the curve $\theta_+(\alpha)$; symbols indicate the mean and variance (over unsuccessful runs) of the fraction of variables assigned before a contradiction is detected, from numerical simulations on 800 formulae of size $N = 2 \times 10^4$ variables for each value of α.

4.4. Numerical experiments on BP guided decimation

We present in this section the results of numerical experiments performed with the BP guided decimation algorithm. According to the definitions given in the general setting, these experiments consisted in generating a random xorsat formula (with $J_a = \pm 1$ with probability one-half) and assigning step by step the value of the variables. The variables were assigned in an uniformly random order. Each time a variable is assigned the BP equations (29) are iterated until convergence is reached or a contradiction is detected (that is, a variable i receives at least two contradicting messages $u = +1$ and -1 from the neighboring clauses). As long as no contradiction is found, the value of the next variable to be assigned is drawn according to the BP estimation of its marginal probability. In this simple model this BP marginal is either completely unbiased (when all incoming messages u from the neighboring clauses vanish), in which case the value of the variable is ± 1 with equal probability, or completely biased, and the assignment is nothing more than the validation of an implication of previous choices. A run of this algorithm is successful if it assigns the value of the N variables without encountering any contradiction; the configuration obtained at the end of the process is then a solution of the formula.

In the left panel of figure 5 we present the probability of successful runs, with respect to the choice of the formula and to the randomness in the course of the run (order of the variables and free choices for unbiased marginals). It goes to a finite value in the thermodynamic limit (which can be computed analytically, see below) for $\alpha < \alpha_*$, and to 0 for $\alpha > \alpha_*$. A further piece of information is given in the right part of figure 5 about the number of steps performed by the algorithm (i.e. the number of variables assigned) before it stops. Let us call this random variable $T_{\rm halt}$ and the associated fraction $\theta_{\rm halt} = T_{\rm halt}/N$. We plotted the mean and variance (represented by error bars) of $\theta_{\rm halt}$, computed only on unsuccessful runs. For $\alpha < \alpha_*$ one finds that $\theta_{\rm halt}$ converges in the thermodynamic limit to a non-trivial random variable. In contrast, in the regime where the algorithm fails w.h.p.

J. Stat. Mech. (2009) P09001

(i.e. for $\alpha > \alpha_*$) the variance of $\theta_{\rm halt}$ vanishes at large N (this result was obtained by performing the simulations at various sizes, which is not shown on the plot). In this case $\theta_{\rm halt}$ concentrates around its mean, which is found to coincide with the function $\theta_+(\alpha)$ defined above (see in particular the inset of the right panel of figure 5).

These numerical observations can now be interpreted in the light of the analytical computations performed above, which were mimicking the decimation process using the perfect marginals instead of the BP estimation. The threshold α_* above which the BP guided decimation algorithm fails w.h.p. coincides with the point where the evolution in the (α, θ) phase diagram has to cross the transition lines drawn in figure 3, and in particular to penetrate the region $[\theta_{\rm c}(\alpha), \theta_+(\alpha)]$ where the 2-core of the residual formula only admits a sub-exponential number of solutions when the perfect marginals are used for the decimation. In this case the algorithm is naturally very sensitive to the small mistakes made by the BP algorithm, which destroy the few solutions of the 2-core. The fact that the residual formula is no longer satisfiable remains, however, unnoticed until a fraction $\theta_+(\alpha)$ of the variables have been assigned. At this point the fraction $\phi(\theta)$ of decimated and logically implied variables has a finite discontinuity (see right panel of figure 2), which means that the assignment of a few new variables triggers an avalanche of implications of extensive size. The extensive subgraph of newly implied variables will contain implication cycles which, if some mistakes have been done in the previous assignment steps, will lead to contradictions. More quantitatively, it was underlined above that $\theta_+(\alpha)$ marked the percolation of the sub-formula of length 2 clauses which supports the propagation of the logical implications.

The simplifying symmetries of the xor-satisfiability formulae are such that BP guided decimation is here almost equivalent to the unit clause propagation algorithm with random heuristic (and also to the random pivoting Gaussian elimination algorithm of [35]). The only slight difference between the two lies in the order in which the variables are treated, the logical implications being propagated as soon as they are detected in UCP. In the BP description of the algorithm the implication is effectively taken into account by the propagation of the messages, even if the variable is not explicitly declared as assigned. The behavior of UCP on xor-satisfiability formulae has been studied in [33]: the results we just found are in agreement with the ones of this paper. In particular the phase diagram in figure 3 reproduces the left panel of figure 3 in [33], apart from the difference in the definition of the vertical time axis explained above. The equivalence with UCP allows also the computation of the probability of success in the thermodynamic limit for $\alpha < \alpha_*$. A detailed derivation for satisfiability formulae can be found in [36, 37]; we state here the result without proof:

$$P_{\rm succ} = \exp\left[-\int_0^1 \frac{{\rm d}t}{4(1-t)}\frac{f(t)^2}{1-f(t)}\right] \qquad \text{with } f(t) = \alpha k(k-1)t^{k-2}(1-t). \tag{38}$$

For $k = 3$ this expression can be further simplified:

$$P_{\rm succ} = \exp\left[\frac{3\alpha}{4} - \frac{1}{2}\frac{1}{\sqrt{(\alpha_*/\alpha)-1}}\arctan\left(\frac{1}{\sqrt{(\alpha_*/\alpha)-1}}\right)\right]. \tag{39}$$

This function is plotted as a solid line in the left panel of figure 5 and agrees with the results of the numerical experiments.

5. Application to the SAT ensemble

We turn in this section to the case of random satisfiability formulae. We shall first apply the analytical cavity formalism to this particular model, then present the results of numerical experiments with the BP guided decimation algorithm and confront the two approaches.

5.1. BP equations

Let us begin by expliciting the BP equations (8) for the satisfiability constraints defined in (2). As the variables σ_i are binary the messages $\{\nu_{a\to i}(\sigma_i), \eta_{i\to a}(\sigma_i)\}$ can be parameterized with a single real for each, which we shall denote $\{u_{a\to i}, h_{i\to a}\}$, under the form

$$\nu_{a\to i}(\sigma_i) = \frac{1 - J_i^a \sigma_i \tanh u_{a\to i}}{2}, \qquad \eta_{i\to a}(\sigma_i) = \frac{1 - J_i^a \sigma_i \tanh h_{i\to a}}{2}. \tag{40}$$

As we included the coupling constant J_i^a in these definitions a positive value of, for instance, $h_{i\to a}$ does not indicate a bias of σ_i towards the value $+1$ in the absence of clause a, but rather towards the value $-J_i^a$ that does satisfy a. The message sent by a clause to one of its variables is then found to be

$$u_{a\to i} = f(\{h_{j\to a}\}_{j\in\partial a\setminus i}), \qquad f(h_1,\dots,h_{k-1}) = -\frac{1}{2}\ln\left(1 - \prod_{i=1}^{k-1}\frac{1-\tanh h_i}{2}\right). \tag{41}$$

To give the explicit form of the other set of BP equations it is advisable to introduce some further definitions. We shall call $\partial_+ i$ (resp. $\partial_- i$) the set of clauses in ∂i which are satisfied by $\sigma_i = +1$ (resp. $\sigma_i = -1$), that is $\partial_\sigma i = \{a \in \partial i | J_i^a = -\sigma\}$. Moreover we let $\partial_+ i(a)$ (resp. $\partial_- i(a)$) denote the set of clauses in $\partial i \setminus a$ agreeing (resp. disagreeing) with a on the value i should take. In formulae, $\partial_+ i(a) = \{b \in \partial i \setminus a | J_i^b = J_i^a\}$, $\partial_- i(a) = \{b \in \partial i | J_i^b = -J_i^a\}$. With these notations the message sent by a variable to a clause is

$$h_{i\to a} = \sum_{b\in\partial_+ i(a)} u_{b\to i} - \sum_{b\in\partial_- i(a)} u_{b\to i}, \tag{42}$$

while the marginal probability of a variable i is from (9)

$$\mu_i(\sigma_i) = \frac{1 + \sigma_i \tanh(H_i)}{2}, \qquad H_i = \sum_{a\in\partial_+ i} u_{a\to i} - \sum_{a\in\partial_- i} u_{a\to i}. \tag{43}$$

Finally, when a subset of variables D is fixed to a reference configuration $\underline{\tau}_D$, the BP equations (41) and (42) are complemented with the boundary condition $h_{i\to a}^{\underline{\tau}_D} = -J_i^a \tau_i \infty$ when $i \in D$. Note that in all the numerical implementations of these equations we keep the hyperbolic tangent of the messages u and h which are free from this apparent singularity in the definition of h around a decimated variable.

5.2. The usual cavity method

The replica symmetric version of the cavity method for non-decimated random satisfiability formulae, following the general formalism recalled in section 3.1, corresponds to a probabilistic interpretation of the BP equations (41) and (42) applied to random factor graphs. As the cardinality of $\partial i \setminus a$ converges to a Poisson random variable of parameter αk and the sign J_i^a in the constraints definition are ± 1 with probability one-half, it follows that $|\partial_+ i(a)|$ and $|\partial_- i(a)|$ converge to two independent Poisson random variables with parameter $\alpha k/2$, denoted l_+ and l_- below. One has thus to look for the solution of the distributional equations corresponding to (41) and (42) for the random variables h and u:

$$h \stackrel{\mathrm{d}}{=} \sum_{i=1}^{l_+} u_i - \sum_{i=1}^{l_-} v_i, \qquad u \stackrel{\mathrm{d}}{=} f(h_1, \ldots, h_{k-1}). \tag{44}$$

In these equations $h_1, \ldots, h_{k-1}$ are independent copies of h and $u_1, \ldots, u_{l_+}$ and $v_1, \ldots, v_{l_-}$ are independent copies of u.

A numerical determination of the fixed point distributions solutions of these equations can be achieved by the population dynamics method, revived in this context by [32]. This consists in representing the random variable u (resp. h) by a sample of $\mathcal{N}$ elements $u_1, \ldots, u_{\mathcal{N}}$ (resp. $h_1, \ldots, h_{\mathcal{N}}$). The sample representing h is initialized arbitrarily, for instance $h_j = 0$ for all $j \in [1, \mathcal{N}]$. Then the two samples are updated alternatively as follows. A new sample representing u is obtained from the representation of h by, independently for each $j \in [1, \mathcal{N}]$:

- drawing $k-1$ indices $i_1, \ldots, i_{k-1}$ independently, uniformly in $[1, \mathcal{N}]$;
- setting $u_j = f(h_{i_1}, \ldots, h_{i_{k-1}})$.

Subsequently the sample of h is updated, for each j, by

- drawing l_+ and l_-, two Poisson random variables of mean $\alpha k/2$;
- drawing $l_+ + l_-$ indices $i_1^+, \ldots, i_{l_+}^+, i_1^-, \ldots, i_{l_-}^-$ independently, uniformly in $[1, \mathcal{N}]$;
- setting $h_j = \sum_{n=1}^{l_+} u_{i_n^+} - \sum_{n=1}^{l_-} u_{i_n^-}$.

The replica symmetric description of the solution space of random satisfiability formulae (first obtained with replica computations in [38]) is only valid for low enough values of α. The 1RSB analysis at $m = 1$, described in generic terms in section 3.1, has been performed on the satisfiability ensemble in [11, 30]. For $k \geq 4$ one finds a clustering transition at α_{d} and a condensation one at α_{c}, before the satisfiability transition α_{s} determined in [7]–[9] (for instance, $\alpha_{\mathrm{d}} \approx 9.38$, $\alpha_{\mathrm{c}} \approx 9.55$ and $\alpha_{\mathrm{s}} \approx 9.93$ for $k = 4$). In the intermediate regime $[\alpha_{\mathrm{d}}, \alpha_{\mathrm{c}}]$ the complexity of the relevant clusters is positive and vanishes at α_{c}, a point beyond which most of the solutions are contained in a sub-exponential number of clusters. The value $k = 3$ happens to be a particular case for which the intermediate regime with a positive complexity is absent, which we shall not consider in the following.

J. Stat. Mech. (2009) P09001

5.3. The computation of $\omega(\theta)$

Let us now apply the formalism of section 3.2 to an ensemble of decimated random satisfiability formulae. Using the parameterization (40) of the messages the random variables $(\eta, \widetilde{\eta}^{\tau})_{\ell}$, $(\eta, \eta^{\tau})_{\ell}$ and $(\nu, \nu^{\tau})_{\ell}$ becomes random pairs of reals, respectively $(h, \widetilde{h}^{\tau})_{\ell}$, $(h, h^{\tau})_{\ell}$ and $(u, u^{\tau})_{\ell}$ for $\tau = \pm 1$. In the same spirit as we include the coupling constants in the definitions (40) of the fields u and h, we also 'gauge' the definition of these random variables such that u^{+} (resp. u^{-}) corresponds to the message $u_{a\to i}^{\underline{\tau}_D}$ where $\underline{\tau}_D$ is drawn conditional on τ_i satisfying (resp. not satisfying) clause a. The recursion equations (18)–(20) then are

$$\begin{aligned}(h, \widetilde{h}^{\tau})_{\ell} &\stackrel{\mathrm{d}}{=} \begin{cases} (h, h^{\tau})_{\ell} & \text{with probability } 1-\theta \\ (h, \tau\infty) & \text{otherwise,} \end{cases} \\ (h, h^{\tau})_{\ell} &\stackrel{\mathrm{d}}{=} \left(\sum_{i=1}^{l_+} u_i - \sum_{i=1}^{l_-} v_i, \sum_{i=1}^{l_+} u_i^{\tau} - \sum_{i=1}^{l_-} v_i^{-\tau} \right), \end{aligned} \tag{45}$$

where $l_{\pm}$ are two independent Poisson random variables of parameter $\alpha k/2$ and the (u_i, u_i^{τ}) and (v, v_i^{τ}) are independent copies of $(u, u^{\tau})_{\ell}$. Finally equation (20) translates into

$$(u, u^{\tau})_{\ell+1} \stackrel{\mathrm{d}}{=} \left(f(h_1, \ldots, h_{k-1}), f(\widetilde{h}_1^{\tau_1}, \ldots, \widetilde{h}_{k-1}^{\tau_{k-1}}) \right), \tag{46}$$

where the configuration of the variables $\tau_1, \ldots, \tau_{k-1}$ is drawn with one of the two following probability laws according to the value of τ:

$$\mathrm{Prob}[\tau_1, \ldots, \tau_{k-1} | \tau = +, h_1, \ldots, h_{k-1}] = \prod_{i=1}^{k-1} \frac{1 + \tau_i \tanh h_i}{2}, \tag{47}$$

or

$$\begin{aligned}\mathrm{Prob}[\tau_1, \ldots, \tau_{k-1} | \tau = -, h_1, \ldots, h_{k-1}] &= \frac{1 - \mathbb{I}(\tau_1 = \cdots = \tau_{k-1} = -1)}{1 - \prod_{i=1}^{k-1}(1 - \tanh h_i)/2} \\ &\times \prod_{i=1}^{k-1} \frac{1 + \tau_i \tanh h_i}{2}. \end{aligned} \tag{48}$$

The fields h_i are the same for the computation of u in (46) and in the probability law of the τ_is expressed in (47) and (48). The two initial conditions correspond to $(h, h^{\tau})_{\ell=0} \stackrel{\mathrm{d}}{=} (h, h)$ for the initialization called I_0, and $(h, h^{\tau})_{\ell=0} \stackrel{\mathrm{d}}{=} (h, \tau\infty)$ for I_1. Finally the average entropy of the decimated random formulae is from (22)

$$\begin{aligned}\omega = -\alpha k(1-\theta)\mathbb{E}\left[\sum_{\tau} \frac{1 + \tau \tanh(u+h)}{2} \ln\left(\frac{1 + \tanh u^{\tau} \tanh h^{\tau}}{2} \right) \right] \\ + \alpha \mathbb{E}\left[\sum_{\tau_1, \ldots, \tau_k} \frac{1 - \mathbb{I}(\tau_1 = \cdots = \tau_k = -1)}{1 - \prod_{i=1}^{k}(1 - \tanh h_i)/2} \prod_{i=1}^{k} \frac{1 + \tau_i \tanh h_i}{2} \right. \\ \left. \times \ln\left(1 - \prod_{i=1}^{k} \frac{1 - \tanh \widetilde{h}_i^{\tau_i}}{2} \right) \right] \end{aligned}$$

J. Stat. Mech. (2009) P09001

$$+ (1-\theta)\mathbb{E}\Bigg[\sum_{\tau}\frac{1+\tau\tanh(\sum_{i=1}^{l_+}u_i-\sum_{i=1}^{l_-}v_i)}{2} \times \ln\Bigg(\sum_{\sigma}\prod_{i=1}^{l_+}\frac{1+\sigma\tanh u_i^{\tau}}{2}\prod_{i=1}^{l_-}\frac{1-\sigma\tanh v_i^{-\tau}}{2}\Bigg)\Bigg]. \tag{49}$$

A numerical determination of the distribution of the random variables $(h,h^{\tau})_{\ell}$ and $(u,u^{\tau})_{\ell}$ can be performed by a population dynamics algorithm. We introduce two populations of $\mathcal{N}$ triplets of reals, $\{(h_i,h_i^+,h_i^-)\}_{i=1}^{\mathcal{N}}$ and $\{(u_i,u_i^+,u_i^-)\}_{i=1}^{\mathcal{N}}$, such that, for instance, the empirical distribution of (h_i,h_i^+) after ℓ steps of the algorithm is a good approximation of the random variable $(h,h^+)_{\ell}$. In the initialization step of the algorithm the h_is are drawn according to the fixed point solution of equation (44), which is obtained from a preliminary RS population dynamics procedure. For the initial condition I_0 (resp. I_1) one sets $h_i^+=h_i^-=h_i$ (resp. $h_i^{\pm}=\pm\infty$) for all $i\in[1,\mathcal{N}]$. Then the following two kinds of updates are iterated ℓ times. A new sample of $\{(u_i,u_i^+,u_i^-)\}_{i=1}^{\mathcal{N}}$ is obtained by, independently for each $j\in[1,\mathcal{N}]$:

- drawing $k-1$ indices $i_1,\dots,i_{k-1}$ independently, uniformly in $[1,\mathcal{N}]$;
- setting $u_j=f(h_{i_1},\dots,h_{i_{k-1}})$;
- independently for $n=1,\dots,k-1$;
  - with probability θ set $\widetilde{h}_n^+=+\infty$ and $\widetilde{h}_n^-=-\infty$;
  - otherwise set $\widetilde{h}_n^+=h_{i_n}^+$, and $\widetilde{h}_n^-=h_{i_n}^-$;
- generating a configuration $\tau_1,\dots,\tau_{k-1}$ from the law $\mathrm{Prob}[\tau_1,\dots,\tau_{k-1}|\tau=+,h_{i_1},\dots,h_{i_{k-1}}]$ defined in equation (47);
- setting $u_j^+=f(\widetilde{h}_1^{\tau_1},\dots,\widetilde{h}_{k-1}^{\tau_{k-1}})$;
- generating a configuration $\tau_1,\dots,\tau_{k-1}$ from the law $\mathrm{Prob}[\tau_1,\dots,\tau_{k-1}|\tau=-,h_{i_1},\dots,h_{i_{k-1}}]$ defined in equation (48);
- setting $u_j^-=f(\widetilde{h}_1^{\tau_1},\dots,\widetilde{h}_{k-1}^{\tau_{k-1}})$.

Subsequently the sample of $\{(h_i,h_i^+,h_i^-)\}_{i=1}^{\mathcal{N}}$ is updated, for each j, by

- drawing l_+ and l_-, two Poisson random variables of mean $\alpha k/2$;
- drawing l_++l_- indices $i_1^+,\dots,i_{l_+}^+,i_1^-,\dots,i_{l_-}^-$ independently, uniformly in $[1,\mathcal{N}]$;
- setting $h_j=\sum_{n=1}^{l_+}u_{i_n^+}-\sum_{n=1}^{l_-}u_{i_n^-}$, $h_j^+=\sum_{n=1}^{l_+}u_{i_n^+}^+-\sum_{n=1}^{l_-}u_{i_n^-}^-$ and $h_j^-=\sum_{n=1}^{l_+}u_{i_n^+}^--\sum_{n=1}^{l_-}u_{i_n^-}^+$.

After a large number of these iterations have been performed the determination of the residual entropy (49) is easily obtained: the expectation values can be interpreted as empirical averages over the population.

We have implemented this numerical procedure and performed the computation for various values of α and θ. The results for $k=4$ are as follows. For small enough values of α the large ℓ limit of the recursion relations (45) and (46) is found to be independent of the initial condition I_0 or I_1 used, and the residual entropy density $\omega(\theta)$ is a smoothly decreasing function. This quantity is plotted for $\alpha=8.8$ in the left panel of figure 6.

J. Stat. Mech. (2009) P09001

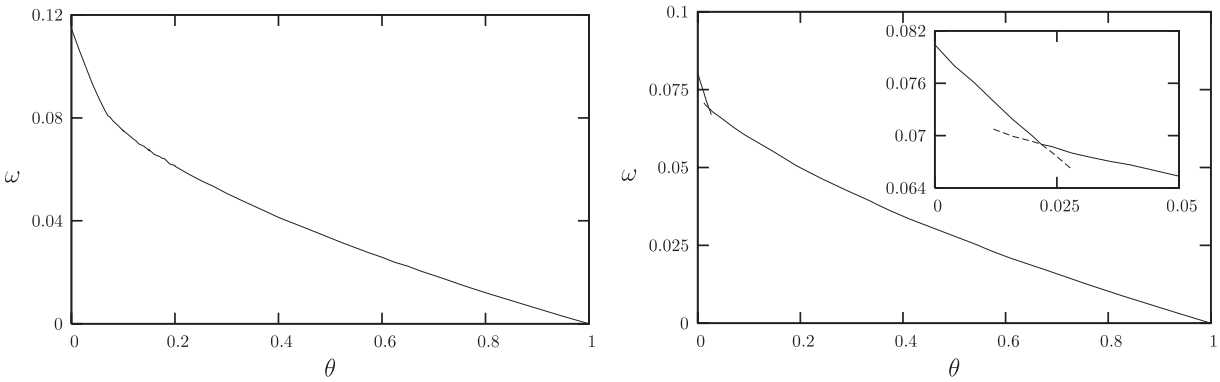

Figure 6. Residual entropy $\omega(\theta)$ of 4-sat random formulae. Left panel: $\alpha = 8.8$, $\omega(\theta)$ is smoothly decreasing. Right panel: $\alpha = 9.3$, the inset is a zoom around the singular point of $\omega(\theta)$.

For larger values of α there appears a regime $\theta \in [\theta_-(\alpha), \theta_+(\alpha)]$ in which the two initial conditions leads to different fixed point solutions of (45) and (46), signaling the presence of non-trivial long-range point-to-set correlations in the decimated formula. The two branches of $\omega(\theta)$ are plotted in the right panel of figure 6 for $\alpha = 9.3$, and are found to cross each other at $\theta_c(\alpha) \in [\theta_-(\alpha), \theta_+(\alpha)]$. For $\theta \in [\theta_-(\alpha), \theta_c(\alpha)]$ the branch with the highest value of ω corresponds to the I_0 initialization, the situation being reversed for $\theta \in [\theta_c(\alpha), \theta_+(\alpha)]$. As explained in the simpler xorsat example, we interpret these results as following from the existence of a positive complexity of relevant clusters in the regime $[\theta_-, \theta_c]$. In this case the highest branch of $\omega(\theta)$ is the total entropy of the decimated formula, while the difference between the two branches is its complexity. In contrast for $[\theta_c, \theta_+]$ the upper branch is the only relevant one; the total entropy is dominated by the sub-exponential number of clusters around a typical reference solution $\underline{\tau}$. The three critical lines are displayed in the (α, θ) of figure 7, which also shows that, as follows from their definitions, $\theta_-(\alpha)$ (resp. $\theta_c(\alpha)$) reaches the horizontal axis $\theta = 0$ at the usual dynamic transition α_d (resp. condensation threshold α_c). We estimated the location of the critical point where $\theta_\pm$ and θ_c merge to be $\alpha_* = 9.05$, $\theta_* = 0.045$, by interpolation of the results obtained for values of α slightly larger.

5.4. The computation of $\phi(\theta)$

We proceed now with the computation of the fraction of logically implied variables, following the lines sketched in section 3.3 (the same results were presented in [22] with a slightly different formulation).

Let us first discuss the warning propagation equations for satisfiability formulae. According to the projection equations (11) one has to identify the situations in which a single value of a variable σ_i is allowed by a BP message. For the messages sent by a clause to a variable this can only happen when the variable is forced to satisfy the clause, i.e. $\nu_{a\to i}(\sigma_i) = \delta_{\sigma_i, -J_i^a}$, in which case we define the WP message to be $\mathfrak{u}_{a\to i} = 1$, otherwise $\mathfrak{u}_{a\to i} = 0$. A message $\eta_{i\to a}$ sent from a variable to a clause can allow both values of variable σ_i, or force it to the value satisfying a ($\eta_{i\to a}(\sigma_i) = \delta_{\sigma_i, -J_i^a}$), or to the value dissatisfying it ($\eta_{i\to a}(\sigma_i) = \delta_{\sigma_i, J_i^a}$). It is only the latter case that shall be propagated by the WP

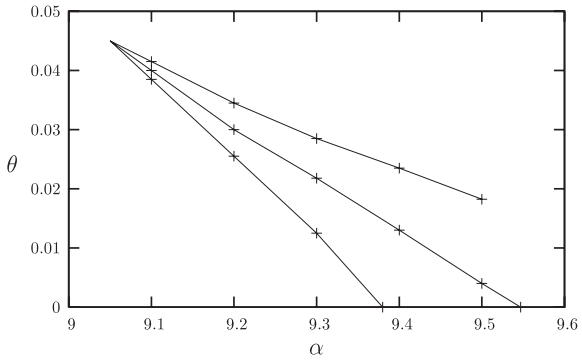

Figure 7. Phase diagram of 4-sat random formulae in the (α, θ) plane, from bottom to top $\theta_-(\alpha)$, $\theta_c(\alpha)$ and $\theta_+(\alpha)$. Symbols result from the population dynamics algorithm, lines are guides to the eyes.

equations. We thus affect the value $\mathfrak{h}_{i\to a} = 0$ in the first two situations and $\mathfrak{h}_{i\to a} = 1$ in the latter. The WP equations are, with these definitions:

$$\mathfrak{u}_{a\to i} = \prod_{j\in\partial a\setminus i} \mathfrak{h}_{j\to a}, \qquad \mathfrak{h}_{i\to a} = 1 - \prod_{b\in\partial_- i(a)} (1-\mathfrak{u}_{b\to i}). \tag{50}$$

For a variable $i \in D$ the boundary condition is $\mathfrak{h}_{i\to a} = \mathbb{I}(\tau_i = J_i^a)$.

In order to compute the average fraction of logically implied variables, within the assumptions of the RS cavity method on the local description of the uniform probability measure $\mu(\cdot)$, we introduce the sequences of random variables $(h, \widetilde{\mathfrak{h}}^\tau)_\ell$, $(h, \mathfrak{h}^\tau)_\ell$ and $(u, \mathfrak{u}^\tau)_\ell$ as defined in section 3.3. It turns out that not all the values of τ have to be considered. Consider, for instance, the random variable $(u, \mathfrak{u}^\tau)_\ell$. Its distribution is by definition the one of $(u_{a\to i}, \mathfrak{u}^{\underline{\tau}_D}_{a\to i})$, in the random tree model of depth ℓ, rooted at variable i which appears solely in the clause a. Depending on τ the reference configuration $\underline{\tau}$ is drawn conditional on τ_i either satisfying (if $\tau = +1$) or not satisfying ($\tau = -1$) the constraint a. In the latter case $\mathfrak{u}^{\underline{\tau}_D}_{a\to i}$ is necessarily equal to 0: at least one of the variables in $\partial a \setminus i$ must satisfy a in $\underline{\tau}$, and this variable cannot be forced to its opposite value by $\underline{\tau}_D$. We can hence restrict our attention to $(u, \mathfrak{u}^+)_\ell$, which is found to obey

$$\begin{aligned} (u, \mathfrak{u}^+)_{\ell+1} &\stackrel{\mathrm{d}}{=} (f(h_1, \ldots, h_{k-1}), \zeta\, \widetilde{\mathfrak{h}}_1^- \cdots \widetilde{\mathfrak{h}}_{k-1}^-), \\ \zeta &\stackrel{\mathrm{d}}{=} \begin{cases} 1 & \text{with probability } \prod_{i=1}^{k-1} \frac{1-\tanh h_i}{2} \\ 0 & \text{otherwise.} \end{cases} \end{aligned} \tag{51}$$

The probability that the random variable ζ equals 1 is indeed the probability that, conditional on τ_i satisfying the root clause a, the configuration of the $k-1$ other variables in $\partial a \setminus i$ are drawn to the values unsatisfying a.

For similar reasons the right-hand side of this equation does not depend on $(h, \widetilde{\mathfrak{h}}^+)$ and we can complete this equation with the recursion on $(h, \widetilde{\mathfrak{h}}^-)$ and $(h, \mathfrak{h}^-)$, which is

$$\begin{aligned} (h, \widetilde{\mathfrak{h}}^-)_\ell &\stackrel{\mathrm{d}}{=} \begin{cases} (h, \mathfrak{h}^-)_\ell & \text{with probability } 1-\theta \\ (h, 1) & \text{otherwise,} \end{cases} \\ (h, \mathfrak{h}^-)_\ell &\stackrel{\mathrm{d}}{=} \left(\sum_{i=1}^{l_+} u_i - \sum_{i=1}^{l_-} v_i, 1 - \prod_{i=1}^{l_-} (1-\mathfrak{v}_i^+) \right), \end{aligned} \tag{52}$$

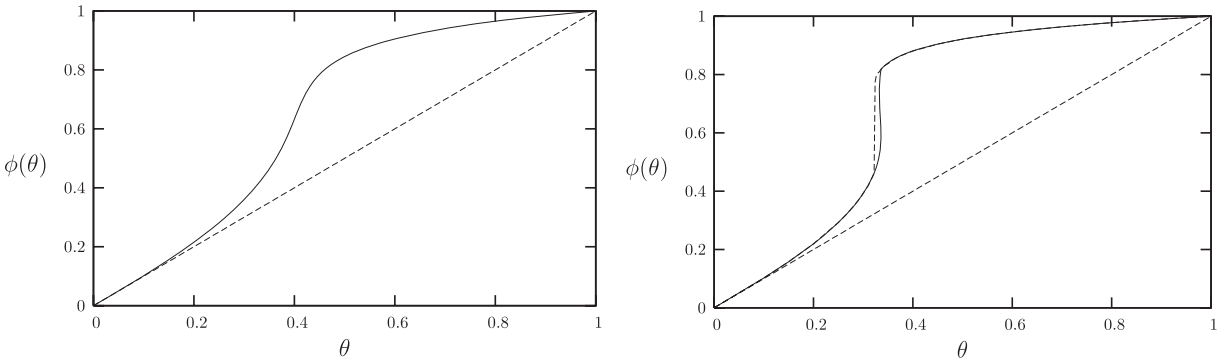

Figure 8. Fraction $\phi(\theta)$ of assigned and logically implied variables in 4-sat random formulae; left: $\alpha = 7.0$, right: $\alpha = 8.4$.

where as usual $l_\pm$ are two Poisson random variables of parameter $\alpha k/2$ and the $(u_i, \mathfrak{u}_i^+)$ and $(v_i, \mathfrak{v}_i^+)$ are independent copies of $(u, \mathfrak{u}^+)_\ell$. Finally the average fraction of either decimated or directly implied variables can be obtained as

$$\phi(\theta) = \mathbb{E}[(1 - \tanh h)\widetilde{\mathfrak{h}}^-]. \tag{53}$$

These recursion equations can be solved numerically using the same kind of population dynamics as explained above, updating in turns populations of pairs $\{(h_i, \mathfrak{h}_i^-)\}_{i=1}^{\mathcal{N}}$ and $\{(u_i, \mathfrak{u}_i^-)\}_{i=1}^{\mathcal{N}}$. The two kinds of initial conditions already discussed correspond here to $(h, \mathfrak{h}^-)_{\ell=0} \stackrel{\mathrm{d}}{=} (h, 0)$ for I_0, and $(h, \mathfrak{h}^-)_{\ell=0} \stackrel{\mathrm{d}}{=} (h, 1)$ for I_1.

This numerical resolution leads to the following results for $k = 4$. At small enough values of α the two initial conditions lead to the same large ℓ limit and the function $\phi(\theta)$ is smoothly increasing (see left panel of figure 8). For larger values of α there exists a range of parameter $[\theta'_-(\alpha), \theta'_+(\alpha)]$ where the quantity (53), computed from the initial condition I_1, is strictly greater than the one reached from I_0. In this coexistence regime we shall call $\psi(\theta)$, in analogy with the notations used for the xorsat model, the upper branch obtained from I_1, see, for instance, the right panel of figure 8. The function ϕ (resp. ψ) is discontinuous at θ'_+ (resp. θ'_-). These two thresholds are the two upper curves in the phase diagram of figure 9, which also contains for comparison a repetition of the phase diagram of figure 7. The two regimes for the behavior of $\phi(\theta)$ are separated by the value $\alpha'_* \approx 8.05$.

At this point the reader might be puzzled by the apparent contradiction between these results and those of the previous subsection. Consider indeed some parameters $\alpha > \alpha'_*$ and $\theta \in [\theta'_-(\alpha), \theta'_+(\alpha)]$. We claimed in the previous subsection that the large ℓ limit of the random variable $(h, h^\tau)_\ell$ was independent of the initial condition in $\ell = 0$, whereas we just found that $(h, \mathfrak{h}^-)_\ell$ does depend on it. As the latter variable is a projection of the former, this statement is at first sight paradoxical. This apparent contradiction can, however, be resolved by a closer inspection of the relationship between the two random variables. One has indeed

$$(h, \mathfrak{h}^-)_\ell \stackrel{\mathrm{d}}{=} \lim_{\varepsilon \to 0} (h, \mathbb{I}(\tanh h^- \leq -1 + \varepsilon))_\ell, \tag{54}$$

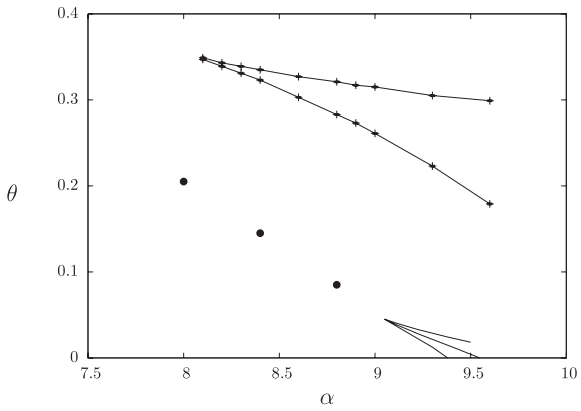

Figure 9. Phase diagram of 4-sat random formulae in the (α, θ) plane. The three lines in the bottom of the figure are those of figure 7, while the upper two are $\theta'_-(\alpha) < \theta'_+(\alpha)$ defined in section 5.4 from the discontinuities of $\phi(\theta)$ and $\psi(\theta)$. The filled circles show the location $\theta_{\max}(\alpha)$ of the slowest convergence of the BP iterations in the BP guided decimation algorithm, see section 5.5.2.

while the two apparently contradictory statements are

$$\lim_{\ell\to\infty}(h,h^-)_\ell^{I_0} \stackrel{\mathrm{d}}{=} \lim_{\ell\to\infty}(h,h^-)_\ell^{I_1}, \qquad \lim_{\ell\to\infty}(h,\mathfrak{h}^-)_\ell^{I_0} \stackrel{\mathrm{d}}{\neq} \lim_{\ell\to\infty}(h,\mathfrak{h}^-)_\ell^{I_1}. \tag{55}$$

The resolution of the paradox relies on the non-commutativity of the limits $\ell \to \infty$ and $\varepsilon \to 0$. More explicitly, under the initialization I_0 there is a positive probability for a field $\tanh h^-$ to have -1 as its large ℓ limit, yet remaining strictly superior to -1 as long as ℓ is finite. If the limit $\varepsilon \to 0$ is taken before $\ell \to \infty$ these fields do not participate in $\mathfrak{h}^-$, which is thus found to be smaller in the initialization I_0 with respect to I_1. Yet if the limit $\ell \to \infty$ is performed first this positive fraction of the fields $\tanh h^-$ (with initialization I_0) reach their limit -1, hence making possible the first statement of equation (55). We checked explicitly this phenomenon by constructing a coupling of the two initializations and solved it with the population dynamics algorithm.

5.5. Numerical experiments on BP guided decimation

We have run the belief propagation guided decimation algorithm for many random 4-sat formulae. The sizes of the formulae studied are $N = 10^3, 3\times10^3, 10^4, 3\times10^4$, with α varying between 6.0 and 9.2. The number of formulae analyzed varies with α, but it is always larger than 2000 for $N = 10^3$, larger than 1200 for $N = 3\times10^3$, larger than 400 for $N = 10^4$ and between 360 and 25 (increasing α) for $N = 3\times10^4$.

5.5.1. Details on the practical implementation. Some technical details about the numerical implementation of the BP guided decimation algorithm were given in [22] (see also appendix A of [30] for details of the representation of BP messages and probabilities). The main numerical bottleneck in applying the BP guided decimation algorithm is the convergence of the iterative method for solving the BP equations, described in section 2.4. This iterative scheme is known to be a fast way of finding a fixed point of the BP equations,

although sometimes it may not converge. Lack of convergence may be due to different reasons: in case long-range correlations develop, multiple BP fixed points appear and the convergence of BP to one of these fixed points cannot be guaranteed; on the other hand, when a single BP fixed point exists, convergence problems can be typically cured by the use of a damping term [20]. In all our numerical simulations we have used a damping term of intensity 0.1, that is, when we update a message `x`, we do not assign to it directly the new value `xNew`, but rather the weighted sum $0.9 * \texttt{xNew} + 0.1 * \texttt{x}$. We have verified that, under these conditions, the convergence (if any) is always exponentially fast in the number of iterative steps (although sometimes with an exponent very small). Because of the exponentially fast convergence, our arbitrary choice of considering BP equations solved when the maximal change in any BP message is below 10^{-4} turns out to be very reasonable: indeed an accuracy of 10^{-8} can be reached by simply doubling the running time. Anyhow, in order to avoid entering a never-ending loop we have also fixed a maximum number of iterations equal to 1000; when this limit is reached, non-converged BP messages are used to compute marginals and to proceed with the decimation.

The last comment about technical issues regards the initialization of BP messages before the iterative solving procedure is applied. In the beginning, when the formula is still not decimated, BP messages are initialized in a random way assigning to each $\tanh(h_{i\to a})$ a random value uniformly distributed between -0.2 and 0.2. After each variable decimation, one can choose to keep the BP messages obtained from the last iterative procedure or to re-initialize them along the same random way described above. In principle, if a single BP fixed point exists and if this is reached by the iterative method, then the starting point should be irrelevant. Moreover, one would expect that the BP fixed points of two formulae differing in just a variable are very close, and that starting from the one already reached should help convergence (with respect to starting from random messages). This intuition turns out to be wrong. We have strong numerical evidence that a random re-initialization of BP messages after each decimation strongly enhances the performance of the algorithm. A possible explanation is the following. Our numerical procedure does not produce a perfect estimation of the marginal probabilities (in particular when the stopping criterion used is the maximal number of iterations); if messages are not re-initialized small errors may easily accumulate in the same direction, while a random re-initialization of BP messages results in a partial neutralization of these errors.

5.5.2. Algorithm performance and convergence probabilities. As a first result, we show in figure 10 the success probability for the BP guided decimation algorithm, i.e. the fraction of formulae which have been solved by this algorithm. The numerical data clearly point to an algorithmic threshold $\alpha_{\rm a}$ very close to the theoretical prediction of the point $\alpha_* = 9.05$ (marked by a vertical line in figure 10) above which phase transitions occur in the thermodynamic properties of the decimated ensemble of random formulae. For $\alpha < \alpha_{\rm a}$ a large N formula is solved with positive probability by the BP guided decimation algorithm. The appearance of a jump in the function $\phi(\theta)$ at $\alpha \simeq 8.1$ (see below for a more detailed analysis of $\phi(\theta)$), with a consequent avalanche of directly implied variables during the decimation of formulae with $\alpha > 8.1$, does not have any visible effect on the success probability. This phenomenon has, however, a trace in the random variable $\theta_{\rm halt}$, which is the fraction of variables assigned before the discovery of a contradiction during the unsuccessful runs. The distribution of this random variable is shown in figure 11 for

doi:10.1088/1742-5468/2009/09/P09001

J. Stat. Mech. (2009) P09001

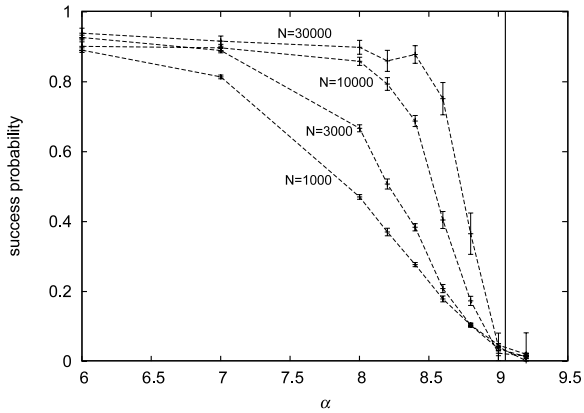

Figure 10. Success probability of the BP guided decimation algorithm as a function of α for random 4-sat formulae of various sizes. The vertical line marks the value $\alpha_* = 9.05$ beyond which the analytical computations predicted a condensation transition in the residual entropy $\omega(\theta)$.

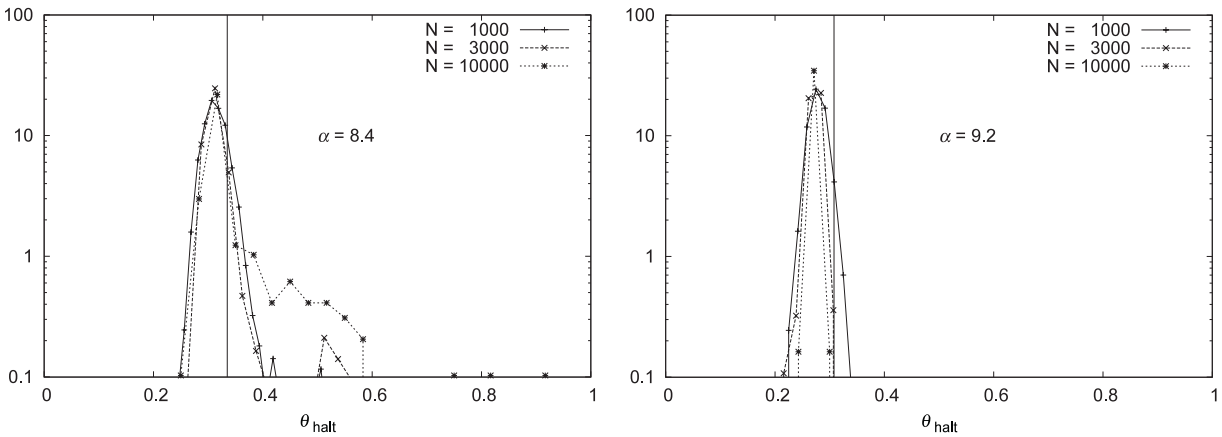

Figure 11. Distribution of the halting time of the BP guided decimation algorithm on 4-sat random formulae with $\alpha = 8.4$ (left panel) and $\alpha = 9.2$ (right panel). Vertical lines show the value of $\theta'_+(\alpha)$ where $\phi(\theta)$ is discontinuous.

two values of α (below and above α_a). One can see a maximum in this distribution for values of θ slightly smaller than $\theta'_+(\alpha)$, the point of discontinuity of $\phi(\theta)$.

In the following we are going to present data only in the region $\alpha < \alpha_a$. In order to reduce finite size effects we will concentrate only on formulae which have been actually solved by our algorithm. The study of the convergence probability and of the average convergence time for the iterative method used to solve the BP equations provides very useful information, as it allows us to identify the most difficult formulae, which should appear close to the threshold. In figure 12 we show both the probability that the BP fixed point is not reached after 1000 iterations (upper panels) and the average number of iterations required to converge (lower panels). Non-converged instances count with 1000 in the average. Four values of α are shown (from left to right), $\alpha = 7.0$, 8.0, 8.4 and 8.8, and $\theta > 0.5$ is not shown since in that region nothing of interest takes place.

J. Stat. Mech. (2009) P09001

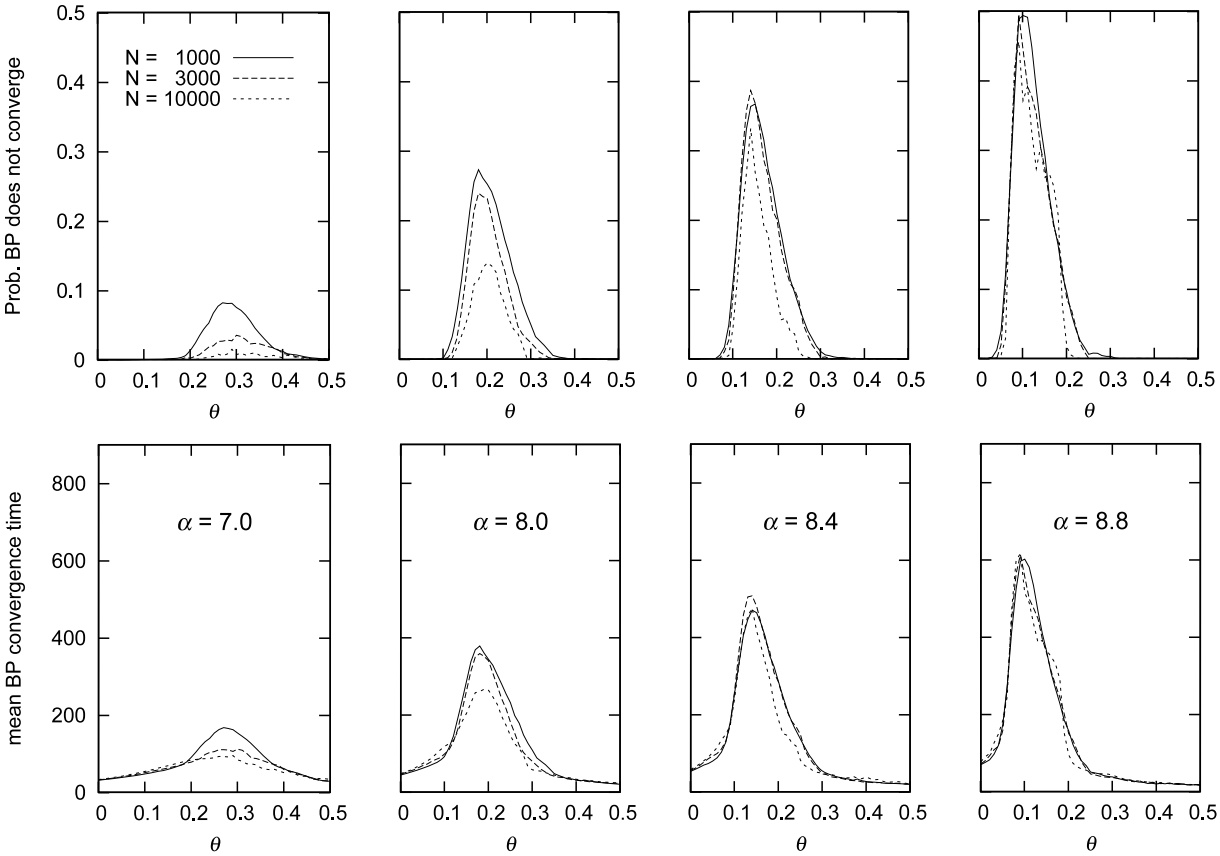

Figure 12. Probability of non-convergence in 1000 iterations (upper panels) and average number of iterations required to reach convergence (lower panels) for the BP part of the BP guided decimation algorithm, as a function of the fraction θ of decimated variables. From left to right $\alpha = 7$, $\alpha = 8$, $\alpha = 8.4$ and $\alpha = 8.8$.

We see that, for small values of α, by increasing the size of formulae the probability that BP does not converge in 1000 steps reduces considerably, thus suggesting that in the large N limit the typical running time of BP is below 1000 for any θ value. In contrast, for larger values of α, the probability that BP does not converge is not varying very much with N and seems to remain positive even in the large N limit, thus suggesting that the typical number of iterations required to make BP converge is larger than 1000 for some values of θ.

The overall picture we get from figure 12 is very clear. For any α value, the decimation procedure initially produces formulae which are more and more difficult to solve and the running time of BP thus increases with θ. The running time (or equivalently the probability of not converging in a fixed number of iterations) has a maximum at a value $\theta_{\max}(\alpha)$ and then decreases again. By increasing α, $\theta_{\max}(\alpha)$ decreases and the running time at $\theta_{\max}$ increases. It is natural to expect that the maximum running time should diverge at the threshold $\alpha_{\rm a}$; moreover, if one assumes that this phenomenon is related to the critical point α_* marking the end of the (first-order) condensation transition line $\theta_{\rm c}(\alpha)$ for $\alpha > \alpha_*$, one should expect that $\theta_{\max}(\alpha)$ is a precursor of the transition line $\theta_{\rm c}(\alpha)$ in the phase $\alpha < \alpha_*$. The data of $\theta_{\max}(\alpha)$ plotted with filled circles in figure 9 are in agreement

J. Stat. Mech. (2009) P09001

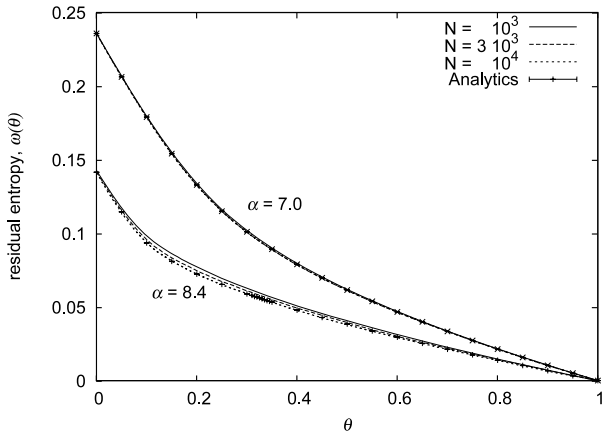

Figure 13. Residual entropy as a function of the fraction θ of decimated variables for $\alpha = 7$ (upper curves) and $\alpha = 8.4$ (lower curves). The symbols correspond to the analytical predictions presented in section 5.3, the lines to the numerical simulations of the BP guided decimation algorithm for various sizes.

with this intuition, showing in particular that $\theta_{\max}(\alpha)$ reaches values very close to θ_* for the largest values of α the algorithm is able to handle.

5.5.3. Entropy of decimated formulae. We measured the entropy of decimated formulae along the execution of the BP guided decimation algorithm, using the Bethe approximation stated in (10). When comparing these results with the analytic prediction presented in section 5.3, a lot of care is required in dealing with cases where BP did not converge to a fixed point. Indeed in these cases the marginal probabilities do not satisfy the consistency constraints and the resulting value for the entropy may be quite far from the correct one. In order to avoid this problem we have adopted a drastic, but safe, approach: we take the average over only those formulae for which the algorithm always converged before reaching the 1000 iterations limit. A possible criticism to this approach is that we are taking the average over the simplest formulae, thus obtaining a biased estimate for the residual entropy. If this criticism is well founded we should observe a dependence of the average residual entropy upon the value of the maximal number of iterations. Actually we do not observe any variation by doubling the maximal number of iterations.

In figure 13 we plot the residual entropy as a function of θ for three different sizes (lines) together with the analytical predictions of section 5.3 (points with errors). For $\alpha = 7.0$, even the $N = 10^3$ data are almost superimposed on the analytical result. In contrast, for $\alpha = 8.4$ finite size effects are much more evident and only $N = 10^4$ data start to be compatible with the analytical computations in the thermodynamic limit.

It is also worth noticing that the function $\omega(\theta)$ shows its point of maximum curvature close to $\theta_{\max}$. More in general the curvature of $\omega(\theta)$ seems to be somehow related to the typical running time of BP.

5.5.4. Forced variables and multiple WP fixed points. We also keep track of the fraction of logically implied variables in the partially decimated formulae, measuring it with the WP

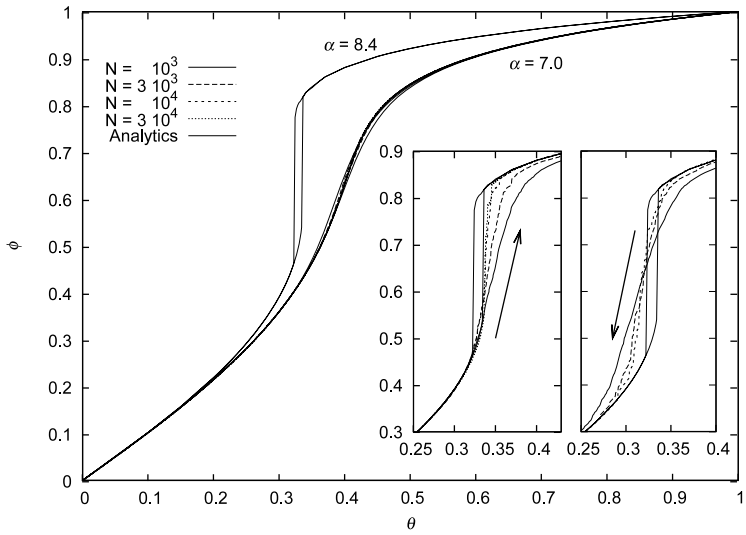

Figure 14. The fraction $\phi(\theta)$ of assigned and logically implied variables for 4-sat formulae of density $\alpha = 7$ and 8.4. The insets detail the results at $\alpha = 8.4$ for the decimation and backward algorithm, see the text for details.

algorithm (equivalent to UCP as explained in section 2.4). In the main panel of figure 14 we plot the function $\phi(\theta)$ computed analytically (solid curves). For $\alpha = 7.0$ the function $\phi(\theta)$ is smoothly increasing and the numerical data follow this curve so nicely that only the $N = 10^3$ data is hardly visible, the rest being perfectly superimposed to the analytical curve. For $\alpha = 8.4$ the function $\phi(\theta)$ is multivalued in the interval $[\theta'_-(\alpha), \theta'_+(\alpha)]$ and the two curves plotted in the main panel correspond to the two branches $\phi(\theta)$ and $\psi(\theta)$ defined in section 5.4. The numerical data for $\alpha = 8.4$ are shown in the insets for clarity. The left inset corresponds to the decimation algorithm (which indeed increases θ during the run). The right inset reports the data gathered while running the *backward* algorithm which works as follows.

After a successful run of the BP guided decimation algorithm we use the solution $\underline{\tau}$ constructed as the reference one, and variables are unfixed one by one in the reverse fixing order: in this way at any θ value the residual formula is exactly the same that the decimation algorithm had to work with. The only differences between the two algorithms are the initial values for the WP and BP messages: in the backward algorithm all messages are set initially according to the solution found in $\theta = 1$, namely $\tanh h_{i\to a} = -J_i^a \tau_i$ and $\mathfrak{h}_{i\to a} = \mathbb{I}(\tau_i = J_i^a)$ for all edges. Then the updates of WP and BP are run as usual for the edges outside the decimated ones. The numerical data shown in the insets suggest that, although finite size effects are huge, in the large N limit the decimation (resp. the backward) algorithm follow the lower branch $\phi(\theta)$ (resp. upper branch $\psi(\theta)$) curve shown in the main panel. As predicted by the analytical computation, for $\alpha > \alpha'_* \simeq 8.05$ the fraction of variables which are frozen (either assigned or directly implied by WP), $\phi(\theta)$, has a hysteresis loop when the number of assigned variables θ is increased and decreased across the interval $[\theta'_-(\alpha), \theta'_+(\alpha)]$, the backward algorithm used when decreasing θ corresponding to the I_1 boundary condition of the infinite tree computation. The hysteresis loop obtained by looking at the WP messages is reported with a full line in figure 15 (where data for $N = 3 \times 10^4$ and $\alpha = 8.4$ are used).

J. Stat. Mech. (2009) P09001

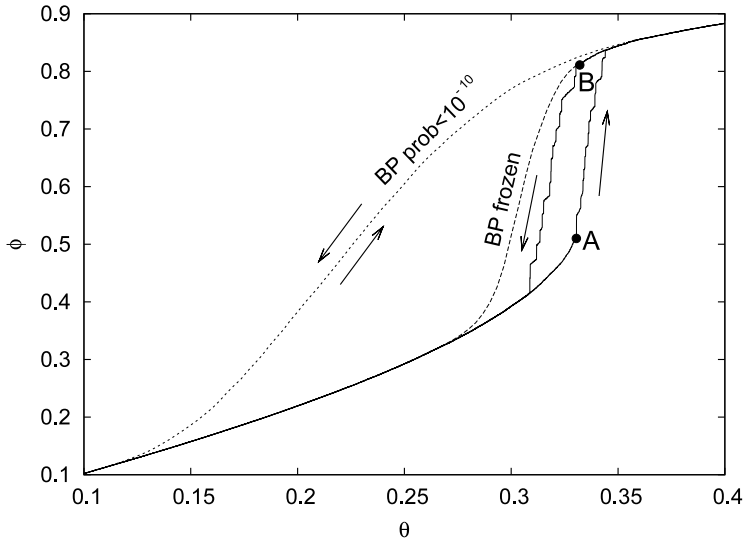

Figure 15. Fraction of frozen variables as a function of θ, for $\alpha = 8.4$. The full lines correspond to the WP results in the decimation and backward algorithm, the dashed lines to the analysis of the BP messages.

The apparent paradox discussed at the end of section 5.4 shows up here again, at the level of the single sample analysis instead of the computation on the infinite tree: the forward and backward procedure allows us to construct two distinct fixed points of the WP equations on the same partially decimated formula (see points A and B in figure 15), while the success probability of the algorithm is still positive for this value of α and the residual entropy of figure 13 has no singularity. In addition to the above discussion on the non-commutativity between the projection from WP to BP and the limit of infinite depth tree/number of iterative updates, it is worth noting that with the initial condition used in the backward algorithm the warning propagation procedure corresponds actually to the whitening construction (see, for instance, [39]), starting from the solution found at the end of the forward algorithm. The interpretation of the number of frozen variables in A and B is thus different: A corresponds to the variables which are logically implied (in the UCP sense) in the partially decimated formula. On the other hand, B counts the number of constrained variables in the core [39] of the reference solution of the partially decimated formula (the one reached by the decimation). The existence of distinct WP fixed points would be a sign of a positive SP complexity if one assumed these fixed points to be exponentially numerous. Even when this assumption is correct it does, however, not lead to a contradiction with the nonexistence of thermodynamically relevant clusters or long-range correlations as defined in (23). The former corresponds indeed to a 1RSB computation with Parisi parameter $m = 0$ and the latter to $m = 1$. We checked indeed that computing the residual entropy with the backward algorithm yields the same result as with the decimation one, whereas in the presence of an extensive thermodynamical complexity we would have obtained only the contribution from the internal entropy of the cluster around the reference solution.

More indications come from the study of BP fixed point messages. In figure 15 we plot the fraction of variables that receive BP messages forcing it to a unique value

(BP frozen) and the fraction of variables which are extremely biased under the BP messages, i.e. the less probable value has a probability smaller than 10^{-10} (BP prob $< 10^{-10}$). The fraction of BP frozen variables again shows hysteresis: in the decimation algorithm BP frozen variables perfectly coincide with WP frozen variables, while in the backward algorithm BP frozen variables are a little more than WP frozen variables because of the numerical impossibility of keeping marginals arbitrarily small. What is more interesting is that the fraction of extremely biased variables (BP prob $< 10^{-10}$) does not show any sign of hysteresis: the same smooth curve is followed by the decimation algorithm as well as by the backward algorithm. This observation suggests that BP marginals obtained by the two algorithms are exactly the same, except for (almost) completely frozen variables.

5.6. Large k behavior

We have seen that the behavior of the ensemble of decimated random formulae is richer in the satisfiability case than for xor-satisfiability, with in particular the appearance of two sets of critical lines, one describing the thermodynamic properties, $\omega(\theta)$, and the other the singularities of the logical implications, $\phi(\theta)$. The common wisdom is, however, that when the length k of the clauses gets large the satisfiability model gets simpler, notably allowing some tight rigorous results [40]–[43], and in fact becomes very similar to the xor-satisfiability one. We shall hence briefly discuss now the large k limit of our results for the decimated ensembles.

Let us begin with the xor-satisfiability case; the results of section 4 having an explicit form, they can be easily turned in asymptotic expansions for large k. For instance, the threshold α_* given in equation (33) is found to behave as $(e/k)(1 + O(k^{-1}))$, while the clustering threshold $\alpha_{\rm d}$ is $(\ln k/k)(1 + O(\ln\ln k/\ln k))$ and the condensation threshold $\alpha_{\rm c}$ go to 1 in the large k limit. One can also study the behavior of the transition line $\theta_{\rm c}(\alpha)$ in the last of these three asymptotic scales (i.e. for α constant with respect to k). After a short computation, which consists in expanding the fixed point solutions $\phi(\theta)$ and $\psi(\theta)$ of equation (32), and the associated entropies (35), one finds that the two leading orders are $\theta_{\rm c}(\alpha) \sim 1 - \alpha - \alpha {\rm e}^{-\alpha k}$.

Large k asymptotic expansions of the non-decimated ($\theta = 0$) ensemble of k-satisfiability were performed for the clustering and condensation thresholds in [30] and in [9] for the satisfiability one. The leading order of the clustering threshold is $2^k(\ln k/k)$, while both condensation and satisfiability occur for values of α around $2^k \ln 2$ (see [9, 30] for the subleading corrections, which are different for $\alpha_{\rm c}$ and $\alpha_{\rm s}$). Consider now the fraction of decimated or implied variables $\phi(\theta)$, computed for the satisfiability ensemble according to equations (51) and (52). In the large k limit a crucial simplification occurs thanks to the concentration, at the leading order, of the h and u random variables solutions of the RS equation (44) around 0. From this fact it follows easily (compare with equations (30) and (31)) that the functions $\phi(\theta)$ and $\psi(\theta)$, for satisfiability random formulae of clause density α, approach the corresponding functions of xor-satisfiability formulae of clause density $\alpha/2^k$. In consequence this correspondence holds for the lines $\theta'_{\pm}(\alpha)$, and the critical point α'_* of satisfiability is expected to scale as $(e2^k/k)(1+O(k^{-1}))$. The asymptotic study of the thermodynamic lines $\theta_{\pm}(\alpha)$ is slightly more involved because of the continuous nature of the second member of the pair in the random variable (h, h^τ), whereas it can

J. Stat. Mech. (2009) P09001

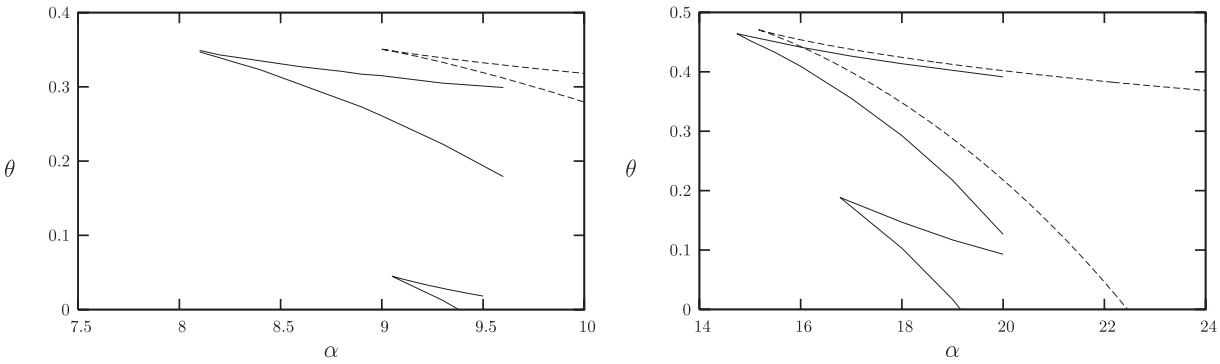

Figure 16. Critical lines $\theta_{\pm}(\alpha)$ and $\theta'_{\pm}(\alpha)$ for k-satisfiability, left: $k = 4$, right: $k = 5$. The dashed lines are the k-xor-satisfiability critical lines, with a rescaling of 2^k on their clause density.

Table 1. Values of the thresholds α_* and α'_*, and the associated fraction of decimated variables θ_* and θ'_*, for random k-satisfiability formulae, $k = 4, 5$.

k	α_*	θ_*	α'_*	θ'_*
4	9.05	0.045	8.05	0.35
5	16.8	0.188	14.7	0.46

only take two values in $(h, \mathfrak{h}^\tau)$. One can, however, notice that a consistent ansatz in the large k limit is to assume $(h, \tanh h^\tau) \approx (h, \tau\mathfrak{h}^\tau)$, that is all nonstrictly forcing messages are approximated as completely unbiased. If this hypothesis is correct the distinction between $\theta_+(\alpha)$ (resp. $\theta_-(\alpha)$, α_*) and $\theta'_+(\alpha)$ (resp. $\theta'_-(\alpha)$, α'_*) should vanish in the large k limit. We have not attempted to obtain a formal proof of this statement but repeated the determination of the satisfiability phase diagram for $k = 5$. The results presented in figure 16 (and in table 1 for the values of the thresholds) confirms the intuition stated above. The two sets of critical lines are much closer for $k = 5$ than $k = 4$, and also in better agreement with the xor-satisfiability values (dashed lines, with a rescaling factor of 2^k on the α axis). Finally an expansion of the residual entropy at the leading order leads us to conjecture that asymptotically $\theta_c(\alpha) \sim 1 - (\alpha/\alpha_c(k))$, as obtained explicitly in the xor-satisfiability case. The data of $\theta_c(\alpha)$ obtained by the population dynamics algorithm for $k = 5$ (not shown) are already in good agreement with this asymptotic form.

We have also performed BP guided decimation simulations for $k = 5$ and found an algorithmic threshold α_a between 16 and 16.5. This range is clearly above the appearance of the jump in $\phi(\theta)$, thus confirming the results presented in section 5.5.2 for $k = 4$, but it is also a little bit below the thermodynamic triple point (mainly because, for $k = 5$, the constraint on the maximum number of iterations produces a more drastic effect).

6. Conclusions

In this paper we have introduced analytical tools that allow computations similar to the Franz–Parisi quenched potential for diluted mean-field systems. We have precisely defined ensembles of partially decimated random CSP and refined their analytical description

J. Stat. Mech. (2009) P09001

initiated in [22]. These methods have been applied to xor-satisfiability random formulae, putting known results [28, 29, 33, 34] in a slightly different perspective, and to the satisfiability case which presents a much richer phase diagram. These computations are expected to describe the behavior of an hypothetical ideal decimation algorithm based on an oracle able to compute exact marginal probabilities in large graphical models.

We have then confronted these results with the outcomes of extensive simulations of the BP guided decimation algorithm, which is a practical, approximate implementation of the ideal procedure. In the case of xor-satisfiability formulae the interpretation of the comparison is very clear and can, in fact, be confirmed by rigorous calculations. The satisfiability problem is much more difficult; the interpretation of the results of the BP decimation should be based on a precise description of the 'small' errors made by BP, which somehow accumulate along the decimation for large enough values of α and cannot avoid conflicting choices in the decimated variables. Lacking such a refined control of BP we have to turn to a more intuitive explanation, based on the analysis of the ideal algorithm. The algorithmic threshold $\alpha_{\rm a}$ for BP decimation on random 4-satisfiability formulae is found to be very close to the value α_* above which clustering and condensation transitions do occur in the (α, θ) plane. One is thus led to conjecture more generally that the presence of a condensed regime, in which BP is expected to fail because of replica symmetry breaking effects, will coincide with the BP decimation threshold for generic CSP.

It is fair to say that we first found the numerical results reported in this work quite surprising. We initially expected [22] the BP guided decimation algorithm to fail when $\phi(\theta)$ develops a jump, that is when the assignment of a single variable produces an avalanche of $O(N)$ forced variables: in this situation, we expected that a contradiction would be generated with high probability. Our numerical results clearly show this is not the case: the algorithm we have studied is able to fix *at the same time* a finite fraction of variables without entering a contradiction. Moreover, as soon as a thermodynamical condensation transition is taking place, for $\alpha \geq \alpha_*$, the success probability falls down to zero sharply. The most natural explanation for these observations is that the marginals used by the algorithm to fix variables are extremely close to the true marginals for any $\alpha < \alpha_*$ (and this is clearly very good news for the use of BP even very close to the clustering threshold) and not so good above α_*. Still some small differences between BP marginals and true marginals are expected even below α_*, mainly given by $1/N$ corrections to the Bethe approximation [44, 45] and to the precision used to solve BP equations ($\epsilon < 10^{-4}$ in our case). Then, why these small differences between true marginals and BP estimations do not affect the success probability of the algorithm before α_*, while they become relevant above α_*? Maybe because the nature of these errors changes crossing α_*. Roughly speaking, below α_* they have a statistical origin and produce random perturbations of intensity $1/N$ and ϵ (in a sense that should be precised): if these errors are largely uncorrelated, when summing $O(N)$ of these we still get errors of order $1/\sqrt{N}$ and $\epsilon\sqrt{N}$, which are very small. In contrast, above α_*, deviations from true marginals are *systematic*, because of long-range correlations: in this case errors are strongly correlated and summing $O(N)$ of these produces a contradiction with high probability.

Let us also sketch a few possible directions for future research. Apart from the computation of the usual Franz–Parisi potential, the analytical formalism can be adapted to other CSP models besides the satisfiability and xor-satisfiability cases treated here.

For instance, the case of coloring should be relatively easy because of the triviality of the RS description (as in xorsat), and relates to the recent study of [46, 47]. It would also be interesting to perform simulations of the BP decimation algorithm on coloring formulae (this was done in [48] but with a bias in the choice of the decimated variable) and check whether our conjecture on the closeness of $\alpha_{\rm a}$ and α_* holds in this case.

A more rigorous analysis of the BP guided decimation algorithm would be welcome, and should be easier to perform in the large k limit. Until very recently the highest clause densities where algorithms were rigorously shown to succeed in finding solutions were $O(2^k/k)$ [36]. Our rough analysis suggests that the threshold of success for BP guided decimation should be on that scale too, with $\alpha_{\rm a}(k) \sim e2^k/k$. It has been shown very recently in [42] that a polynomial time algorithm can find solutions of formulae with densities up to $2^k \ln k/k$, which correspond to the scale of the dynamic threshold $\alpha_{\rm d}(k)$.

On the algorithmic side many variations on the simplest procedure studied in this paper are more efficient (and notably the survey propagation algorithm [7, 16]), yet seems much more difficult to tackle analytically. A slight modification of the BP decimation algorithm where the order of the assignments is not uniformly random but treats in priority variables with the most biased marginals already changes substantively the highest value of α where formulae are solved with positive probability [11]. Other interesting directions to explore would be more efficient procedures for the resolution of the BP equations (for instance double-loop algorithms [49]), the use of reinforcement strategies [50] instead of explicit decimation, or the coordinated decimation of groups of variables [51].

Acknowledgments

We warmly thank Andrea Montanari with whom part of this work was done and who made important suggestions. We are also grateful to Francesco Zamponi and Lenka Zdeborova for a critical reading of the manuscript. This work has been partially funded by the PHC Galileo program for exchanges between France and Italy.

Appendix: Details on the computations of section 4.3

We present in this appendix some details of the computations discussed in section 4.3. The properties of decimated xorsat formulae will be derived through the analysis of two algorithms which act on the formula and which can be described by the differential equation method [52]–[54]. Two successive steps will be performed: the logical implications of the decimation of a fraction θ of the variables are first drawn with unit propagation. Then the structure of the set of solutions of the reduced formula is analyzed by the leaf removal algorithm. This approach has been developed for k-xorsat formulae in [28, 29] and generalized to arbitrary degree distributions in [33]; we reproduce here their results for the sake of self-containedness.

A.1. Unit propagation

As explained at the beginning of section 4.3 we can assume here the formula to be an unfrustrated ($J_a = 1$) set of $M = \alpha N$ equations of the form (1), each involving k indices chosen uniformly at random among the N variables. A fraction θ of the variables are then set to $+1$, and can then be removed from the clauses where they appeared. Let us

call $R_\kappa = N\rho_\kappa$ the number of clauses with κ non-decimated variables, and $L_l = N\lambda_l$ the number of variables which appear in l clauses (note that a decimated variable does not appear in any clause after the above simplification step). One obtains

$$\lambda_l = (1-\theta)\mathrm{e}^{-\alpha k}\frac{(\alpha k)^l}{l!} + \theta\delta_{l,0}, \tag{A.1}$$

$$\rho_\kappa = \alpha\binom{k}{\kappa}(1-\theta)^\kappa\theta^{k-\kappa} \qquad \text{for } \kappa \le k. \tag{A.2}$$

Consider now the action of the unit propagation algorithm. As long as clauses of length $\kappa = 1$ are present in the formula, one of them is chosen randomly, the single variable it contains is fixed to $+1$, and is then removed from the other clauses it appeared in. The formula obtained after T steps of this procedure is uniformly random, conditional on the values of $\{R_\kappa(T), L_l(T)\}$, so that the analysis of the process amounts to following these random variables. At each time step $T \to T+1$ they vary by a bounded random increment whose distribution depends only on the current values $\{R_\kappa(T), L_l(T)\}$ and not on the previous history of the process. As a consequence the reduced quantities $\rho_\kappa(t) = R_\kappa(T = Nt)/N$ and $\lambda_l(t) = L_l(T = Nt)/N$ concentrate around their average values [52]–[54], solutions of the following set of differential equations:

$$\frac{\mathrm{d}}{\mathrm{d}t}\lambda_l(t) = \delta_{l,0} - \frac{l\lambda_l(t)}{\sum_{l'} l'\lambda_{l'}(t)}, \tag{A.3}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\rho_\kappa(t) = -\delta_{k,1} + \left(\frac{\sum_l l(l-1)\lambda_l(t)}{\sum_l l\lambda_l(t)}\right)\left[\frac{(\kappa+1)\rho_{\kappa+1}(t)}{\sum_{\kappa'}\kappa'\rho_{\kappa'}(t)} - \frac{\kappa\rho_\kappa(t)}{\sum_{\kappa'}\kappa'\rho_{\kappa'}(t)}\right]. \tag{A.4}$$

These expressions arise because the variable selected in the unit clause is present in l clauses with probability proportional to $l\lambda_l(t)$; apart from the unit clause, the $l-1$ other occurrences of the variable take place in clauses of length κ with probability proportional to $\kappa\rho_\kappa(t)$.

In order to solve these equations we introduce the generating function of the initial distribution of degrees of the variables, $\lambda(x) = \sum_l \lambda_l x^l$. Equation (A.3) is solved for any $l \ge 1$ by $\lambda_l(t) = \lambda_l a(t)^l$, where $a(t)$ is the solution of

$$\frac{\mathrm{d}}{\mathrm{d}t}a(t) = -\frac{1}{\lambda'(a(t))}, \qquad \text{with the initial condition } \; a(t=0) = 1. \tag{A.5}$$

One can then insert this expression of $\lambda_l(t)$ in (A.4) and solve to obtain

$$\rho_\kappa(t) = \sum_{\kappa'\ge\kappa}\binom{\kappa'}{\kappa}\rho_{\kappa'}\left(\frac{\lambda'(a(t))}{\lambda'(1)}\right)^\kappa\left(1-\frac{\lambda'(a(t))}{\lambda'(1)}\right)^{\kappa'-\kappa} - \delta_{\kappa,1}\lambda'(a(t))(1-a(t)). \tag{A.6}$$

The differential equations (A.3) and (A.4) only make sense if $\rho_1(t) > 0$: the procedure stops when no more unit clause can be selected, i.e. at the reduced stopping time $t_* = \min\{t : \rho_1(t) = 0\}$.

For the initial degree distribution given in (A.1) one obtains for the derivative of the generating function: $\lambda'(x) = \alpha k(1-\theta)\mathrm{e}^{-\alpha k(1-x)}$, hence by integration of (A.5)

$$a(t) = 1 - \frac{1}{\alpha k}\ln\left(\frac{1-\theta}{1-\theta-t}\right). \tag{A.7}$$

J. Stat. Mech. (2009) P09001

Plugging this result in the expression (A.6) of $\rho_1(t)$, one finds that t_* is the smallest solution of

$$\alpha k(\theta + t)^{k-1} = \ln\left(\frac{1-\theta}{1-\theta-t}\right). \tag{A.8}$$

Defining finally $\phi = \theta + t_*$, one realizes that ϕ is the smallest solution of equation (32): this quantity gives the fraction of variables that are either fixed by the decimation or by the propagation of logical implications. When all these implications are taken into account, the degree distributions of the reduced formula are

$$\lambda_l(t_*) = (1-\phi)\mathrm{e}^{-\alpha k(1-\phi^{k-1})}\frac{(\alpha k(1-\phi^{k-1}))^l}{l!} \qquad \text{for } l \geq 1, \tag{A.9}$$

$$\rho_\kappa(t_*) = \alpha\binom{k}{\kappa}(1-\phi)^\kappa\phi^{k-\kappa} \text{ for } \kappa \geq 2. \tag{A.10}$$

As a consequence the number of non-trivial clauses is

$$M' = N\sum_{\kappa=2}^{k}\rho_\kappa(t_*) = \alpha N\left(1-\phi^k - k(1-\phi)\phi^{k-1}\right). \tag{A.11}$$

A.2. Leaf removal

We shall now analyze the set of solutions of random unfrustrated formulae with degree distributions given by (A.9) and (A.10). Following [28, 29], we consider the action of the leaf removal algorithm on such a hypergraph. Each leaf removal step consists in picking at random one variable of degree 1 (a leaf) and remove the single clause it appeared in. This simplification is repeated until no leaf is left in the graph, which provokes the stopping of the algorithm. There are two possible situations at that point: either all clauses have been removed, or there remains a non-empty 2-core, that is the maximal subgraph of the original formula in which all variables have degree at least 2. In both cases the total entropy is given (in units of $\log 2$) by the initial number of variables minus the initial number of clauses. In the former case the set of solutions is unclustered, while in the latter the solutions are split into an exponential number of clusters. Each cluster corresponds to one solution of the 2-core formula, hence the complexity, i.e. the exponential rate of the number of clusters, is given by the difference between the number of variables and of clauses in the 2-core. Each cluster contains an exponential number of solutions, this internal entropy being associated to the degeneracy arising from the freedom in the choice of the value of the leaf variables when reinserted in the formula in the reverse order with respect to their removal.

The evolution of the degree connectivities during the execution of the leaf removal algorithm can be computed in a very similar manner with respect to the unit propagation case sketched above. With a slight abuse of notation we denote again by $N\lambda_l(t)$ and $N\rho_\kappa(t)$ the average number of variables and constraints of degrees l and κ after Nt steps of the leaf removal algorithm. These quantities obey the following set of differential equations:

$$\frac{\mathrm{d}}{\mathrm{d}t}\rho_\kappa(t) = -\frac{\kappa\rho_\kappa(t)}{\sum_{\kappa'}\kappa'\rho_{\kappa'}(t)}, \tag{A.12}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\lambda_l(t) = -\delta_{l,1} + \delta_{l,0} + \left(\frac{\sum_\kappa \kappa(\kappa-1)\rho_\kappa(t)}{\sum_\kappa \kappa\rho_\kappa(t)}\right)\left[\frac{(l+1)\lambda_{l+1}(t)}{\sum_{l'} l'\lambda_{l'}(t)} - \frac{l\lambda_l(t)}{\sum_{l'} l'\lambda_{l'}(t)}\right]. \tag{A.13}$$

J. Stat. Mech. (2009) P09001

These equations are essentially the same as (A.3) and (A.4) with the role of λ and ρ being exchanged (in the leaf removal algorithm one picks a variable of degree 1; in unit clause propagation it is a clause of degree 1). They can thus be solved with the same technique. Let us define the generating function of the clause lengths at the beginning of the leaf removal, $\rho(x) = \sum_\kappa \rho_\kappa x^\kappa$. At all times $\rho_\kappa(t) = \rho_\kappa b(t)^\kappa$, where $b(t)$ is a solution of

$$\frac{\mathrm{d}}{\mathrm{d}t} b(t) = -\frac{1}{\rho'(b(t))}, \qquad \text{with} \quad b(t=0) = 1. \tag{A.14}$$

The distribution of the variable degrees is then found to be

$$\lambda_l(t) = \sum_{l' \geq l} \binom{l'}{l} \lambda_{l'} \left(\frac{\rho'(b(t))}{\rho'(1)}\right)^l \left(1 - \frac{\rho'(b(t))}{\rho'(1)}\right)^{l'-l} - \delta_{l,1}\rho'(b(t))(1-b(t)). \tag{A.15}$$

The stopping time of the leaf removal algorithm is given by $t_* = \min\{t : \lambda_1(t) = 0\}$. A non-trivial 2-core exists at this stopping time if and only if $b(t_*) > 0$. As the function $b(t)$ is decreasing in time (see equation (A.14)), the value $b_* = b(t_*)$ can also be defined as the largest solution in $[0, 1]$ of

$$\rho'(b_*)(1-b_*) = \frac{\rho'(b_*)}{\rho'(1)} \sum_{l=1}^{\infty} l\lambda_l \left(1 - \frac{\rho'(b_*)}{\rho'(1)}\right)^{l-1}. \tag{A.16}$$

Let us now apply these results to the degree distributions (A.9) and (A.10) of the formula obtained at the end of the unit propagation. These imply the following form of the derivative of the clause length generating function:

$$\rho'(x) = \alpha k(1-\phi)\left((\phi + x(1-\phi))^{k-1} - \phi^{k-1}\right). \tag{A.17}$$

The solution of equation (A.16) is either $b_* = 0$ or the largest strictly positive solution of

$$1 - b_* = \exp[-\alpha k\left(\phi + b_*(1-\phi)\right)^{k-1} - \phi^{k-1}]. \tag{A.18}$$

Defining $b_* = (\psi - \phi)/(1-\phi)$, one realizes that the equation on b_* is equivalent to ψ being the largest solution of equation (32). We have thus justified one of the statements made in section 4.3: when there is only one solution to equation (32), $\psi = \phi$ or in other terms $b_* = 0$. This means that the leaf removal algorithm does not stop before having emptied the complete formula, there is no 2-core and the solution space is not clustered. In contrast the existence of the multiple solutions $\psi > \phi$ corresponds to $b_* > 0$ and hence to the presence of a non-trivial 2-core in the hypergraph of constraints. This latter case corresponds to $\alpha > \alpha_*$ and $\theta \in [\theta_-(\alpha), \theta_+(\alpha)]$.

Let us conclude with the justification of the expressions of the entropy and complexity given in section 4. We have seen that the number of non-implied variables at the end of the unit propagation procedure is $N(1-\phi)$, while the number of non-trivial clauses is given in equation (A.11). In the region of the (α, θ) plane where there is a single solution of equation (32) the entropy (35) is given (in units of $\ln 2$) by the difference between the number of variables and constraints, in agreement with the results of [28, 29]. When a non-empty 2-core subsists at the end of the leaf removal algorithm, one can compute from the solution of (A.12) and (A.13) at the stopping time t_* the number of variables and

J. Stat. Mech. (2009) P09001

clauses in the 2-core:

$$N_{\rm core} = N\left[\psi - \phi - \alpha k(1-\psi)(\psi^{k-1} - \phi^{k-1})\right], \tag{A.19}$$

$$M_{\rm core} = N\alpha\left[\psi^k - \phi^k - k\phi^{k-1}(\psi - \phi)\right]. \tag{A.20}$$

It is then easy to verify that $\widehat{\omega}(\phi) - \widehat{\omega}(\psi) = \ln(2)(N_{\rm core} - M_{\rm core})/N$. When this quantity is positive, that is in the interval $[\theta_-(\alpha), \theta_{\rm c}(\alpha)]$, it is equal to the entropy density associated with the exponential number of solutions of the 2-core. In contrast when it is negative the 2-core with more clauses than variables has only a sub-exponential number of solutions (recall that we conditioned from the beginning on the formula being satisfiable and on the reference configuration unveiled being a solution). One can show in this case that the entropy arising from the variables outside the 2-core is given by $\widehat{\omega}(\psi)$, hence the total entropy of the decimated formula is always given by $\max[\widehat{\omega}(\phi), \widehat{\omega}(\psi)]$, the largest branch as plotted in the right panel of figure 4.

References

[1] Mézard M, Parisi G and Virasoro M A, 1987 *Spin Glass Theory and Beyond* (Singapore: World Scientific)
[2] Talagrand M, 2003 *Spin Glasses: A Challenge for Mathematicians* (Berlin: Springer)
[3] Mézard M and Montanari A, 2009 *Information, Physics, and Computation* (Oxford: Oxford University Press)
[4] Friedgut E, 1999 *J. Am. Math. Soc.* **12** 1017
[5] Franco J, 2001 *Theoret. Comput. Sci.* **265** 147
[6] Dubois O, 2001 *Theor. Comput. Sci.* **265** 187
[7] Mézard M and Zecchina R, 2002 *Phys. Rev.* E **66** 056126
[8] Mézard M, Parisi G and Zecchina R, 2002 *Science* **297** 812
[9] Mertens S, Mézard M and Zecchina R, 2006 *Random Struct. Algorithms* **28** 340
[10] Biroli G, Monasson R and Weigt M, 2000 *Eur. Phys. J.* B **14** 551
[11] Krzakala F, Montanari A, Ricci-Tersenghi F, Semerjian G and Zdeborova L, 2007 *Proc. Nat. Acad. Sci.* **104** 10318
[12] Semerjian G and Monasson R, 2003 *Phys. Rev.* E **67** 066103
[13] Barthel W, Hartmann A and Weigt M, 2003 *Phys. Rev.* E **67** 066104
[14] Seitz S, Alava M and Orponen P, 2005 *J. Stat. Mech.* P06006
[15] Davis M and Putman H, 1960 *J. Assoc. Comput. Mach.* **7** 201
Davis M, Logemann G and Loveland D, 1962 *Commun. ACM* **5** 394
[16] Braunstein A, Mézard M and Zecchina R, 2005 *Random Struct. Algorithms* **27** 201
[17] Feige U, Mossel E and Vilenchik D, 2006 *Proc. RANDOM (Barcelona)*
[18] Coja-Oghlan A, Krivelevich M and Vilenchik D, 2007 *Proc. 13th Int. Conf. on Analysis of Algorithms*
[19] Altarelli F, Monasson R and Zamponi F, 2007 *J. Phys. A: Math. Theor.* **40** 867
[20] Pretti M, 2005 *J. Stat. Mech.* P11008
[21] Aurell E, Gordon U and Kirkpatrick S, 2004 *NIPS: 18th Ann. Conf. on Neural Information Processing Systems*
[22] Montanari A, Ricci-Tersenghi F and Semerjian G, 2007 *Proc. 45th Allerton Conf.* pp 352–9
[23] Zdeborova L and Mézard M, 2008 *J. Stat. Mech.* P12004
[24] Kschischang F, Frey B J and Loeliger H-A, 2001 *IEEE Trans. Inf. Theory* **47** 498
[25] Yedidia J S, Freeman W T and Weiss Y, 2001 *Adv. Neural Inf. Process. Syst.* **13** 689
[26] Franz S and Parisi G, 1995 *J. Physique* I **5** 1401
[27] Garey M R and Johnson D S, 1983 *Computers and Intractability: A Guide to the Theory of NP-Completeness* (San Francisco, CA: Freeman)
[28] Mézard M, Ricci-Tersenghi F and Zecchina R, 2003 *J. Stat. Phys.* **111** 505
[29] Cocco S, Dubois O, Mandler J and Monasson R, 2003 *Phys. Rev. Lett.* **90** 047205
[30] Montanari A, Ricci-Tersenghi F and Semerjian G, 2008 *J. Stat. Mech.* P04004
[31] Mézard M and Montanari A, 2006 *J. Stat. Phys.* **124** 1317
[32] Mézard M and Parisi G, 2001 *Eur. Phys. J.* B **20** 217
[33] Altarelli F, Monasson R and Zamponi F, 2008 *J. Phys.: Conf. Ser.* **95** 012013

doi:10.1088/1742-5468/2009/09/P09001

J. Stat. Mech. (2009) P09001

[34] Measson C, Montanari A and Urbanke R, 2008 *IEEE Trans. Inf. Theory* **54** 5277
[35] Braunstein A, Leone M, Ricci-Tersenghi F and Zecchina R, 2002 *J. Phys. A: Math. Gen.* **35** 7559
[36] Frieze A and Suen S, 1996 *J. Algorithms* **20** 312
[37] Deroulers C and Monasson R, 2006 *Eur. Phys. J.* B **49** 339
[38] Monasson R and Zecchina R, 1997 *Phys. Rev.* E **56** 1357
[39] Maneva E, Mossel E and Wainwright M J, 2007 *J. ACM* **54** 1
[40] Achlioptas D and Peres Y, 2004 *J. Am. Math. Soc.* **17** 947
[41] Achlioptas D and Coja-Oghlan A, 2008 arXiv:0803.2122
[42] Coja-Oghlan A, 2009 arXiv:0902.3583
[43] Montanari A, Restrepo R and Tetali P, 2009 arXiv:0904.2751
[44] Montanari A and Rizzo T, 2005 *J. Stat. Mech.* P10011
[45] Parisi G and Slanina F, 2006 *J. Stat. Mech.* L02003
[46] Krzakala F and Zdeborova L, 2009 arXiv:0901.2130
[47] Zdeborova L and Krzakala F, 2009 arXiv:0902.4185
[48] Zdeborova L and Krzakala F, 2007 *Phys. Rev.* E **76** 031131
[49] Yuille A, 2002 *Neural Comput.* **14** 1691
[50] Chavas J, Furtlehner C, Mézard M and Zecchina R, 2005 *J. Stat. Mech.* P11016
[51] Higuchi S and Mézard M, 2009 arXiv:0903.1621
[52] Kurtz T G, 1970 *J. Appl. Probab.* **7** 49
[53] Wormald N C, 1999 *Lectures on Approximation and Randomized Algorithms* ed M Karonski and H J Proemel (Warsaw: PWN) pp 73–155
[54] Achlioptas D, 2001 *Theor. Comput. Sci.* **265** 159

J. Stat. Mech. (2009) P09001