

OPEN ACCESS

Correlation-based decimation in constraint satisfaction problems

To cite this article: Saburo Higuchi and Marc Mézard 2010 *J. Phys.: Conf. Ser.* **233** 012003

View the [article online](#) for updates and enhancements.

You may also like

- [The decimation scheme for symmetric matrix factorization](#)

Francesco Camilli and Marc Mézard

- [A message-passing approach to random constraint satisfaction problems with growing domains](#)

Chunyan Zhao, Haijun Zhou, Zhiming Zheng et al.

- [Learning from survey propagation: a neural network for MAX-E-3-SAT](#)

Raffaele Marino

The advertisement features the ECS logo and the text "The Electrochemical Society" and "Advancing solid state & electrochemical science & technology". It also includes the text "DISCOVER how sustainability intersects with electrochemistry & solid state science research".

Correlation-based decimation in Constraint Satisfaction Problems

Saburo Higuchi^{1,2} and Marc Mézard¹

¹Laboratoire de Physique Théorique et Modèles Statistiques, CNRS and Université Paris-Sud, Bât 100, 91405 Orsay Cedex, France

²Department of Applied Mathematics and Informatics, Ryukoku University, Otsu, Shiga, 520-2194, Japan

Abstract. We study hard constraint satisfaction problems using some decimation algorithms based on mean-field approximations. The message-passing approach is used to estimate, beside the usual one-variable marginals, the pair correlation functions. The identification of strongly correlated pairs allows to use a new decimation procedure, where the relative orientation of a pair of variables is fixed. We apply this novel decimation to locked occupation problems, a class of hard constraint satisfaction problems where the usual belief-propagation guided decimation performs poorly. The pair-decimation approach provides a significant improvement.

1. Introduction

Recent years have seen an important activity in the use of statistical physics concepts and methods to study discrete optimization problems (for a recent introduction, see [1]). The analysis of random constraint satisfaction problems (CSPs) has shown that the hardest problems are generated for a ratio of constraints-to-variable which is close to the critical ratio where a phase transition is found numerically, separating a “SAT” phase where almost all instances are satisfiable from an “UNSAT” phase where almost all instances are not satisfiable [2, 3, 4, 5]. Methods like the cavity or the replica method, developed in the statistical physics of disordered systems, have proved to be very efficient in locating the phase transition, both in satisfiability [6, 7, 8] and in colouring [9, 10, 11]. Another important result obtained with these methods is the existence of another phase transition, the “clustering transition”, inside the SAT phase: this transition separates a phase where the space of solutions builds a large connected clusters from another SAT phase, close to the SAT-UNSAT threshold, where the space of solutions splits into ergodically separated groups – clusters [12, 6, 7]. Although the replica and cavity methods are not rigorous, some important aspects of their results have been confirmed rigorously: the probability that a random instance is SAT is known to have a sharper and sharper transition when the number of variables increases at a fixed ratio of constraints to variables [13], and the clustering property can be proven for random K -satisfiability formulas when K is large enough [14, 15].

Most interestingly, some of these mean-field based approaches have been turned into efficient algorithms which have the best performance for random satisfiability and colouring in their SAT phase close to the threshold, where the hardest instances are generated. These methods all aim at finding a satisfiable assignment (assuming that there is one). Their starting point is a ‘Boltzmann’-type measure, the uniform measure over all satisfying assignment, and they are

typically based on two main steps. The first one, which has been the topic of many studies, is the use of message-passing methods in order to get an estimate of the marginals of this measure. The archetype of these methods, belief propagation (BP), can be seen as an algorithmic version of the Bethe approximation, used on a given instance [16, 17, 18, 19]. When the system is in a glass phase, the survey propagation (SP) algorithm [7], an algorithmic version of the cavity method, gives very good performance. These message passing methods are used to estimate the marginals of each of the variables, with respect to the uniform measure over all solutions.

The second step makes use of these approximate estimates in order to find an assignment of variables which satisfies all constraints. If message passing would give the exact marginals, this second step would be exact, but in the most interesting problems the marginals obtained by BP or SP are approximate, so that a smart use of the information that they provide is not obvious. So far only two approaches have been used: decimation [7] and reinforcement [20]. Decimation consists in identifying from some criterion the most “polarized” variable (e.g. the one with the smallest entropy), and in fixing it to its most probable value. After this variable has been fixed, one obtains a new, smaller, CSP, to which one can apply recursively the whole procedure (BP-or SP- followed by identifying and fixing the most polarized variable). In reinforcement, one finds from the marginals the most probable value of each variable, and one adds, in the local measure of each variable, an extra bias in this preferred direction. The new CSP therefore has the same number of variables as the original one, but the local measure on each variable has been changed. One iterates this reinforcement procedure until the variables are infinitely polarized. If the algorithm is successful this returns a configuration of variables which satisfies all constraints. These two procedures, BP+decimation and BP+reinforcement, are remarkably efficient in random CSPs like K -satisfiability and graph colouring [10], and perceptron learning [21]. When one approaches the SAT-UNSAT threshold of these problems, a more elaborate version which uses the information on marginals from survey propagation (SP) is more effective [7, 6, 20], and at present the SP-based decimation and reinforcement methods are the most efficient incomplete SAT solvers for random K -satisfiability close to its threshold. Most of the work on these decimation and reinforcement procedures is numerical, as is also our present paper. On the analytical side, a first theoretical approach to the decimation problem has been done recently by [22, 23] who have analyzed the ideal decimation procedure, based on exact marginals.

There are two main reasons for exploring alternatives to the decimation and reinforcement procedures. First of all, it is clear that the estimates of the marginals obtained by message passing contain a lot of information, and it is likely that these two procedures do not exploit it fully. For instance one could think of using this information in order to bias intelligently the stochastic local search methods, or one could couple it to tree-search based solvers. On the second hand, some broad classes of constraint satisfaction problems have been found recently where these procedures perform rather poorly. These are the locked occupation problems (LOPs), a class of CSPs where the set of solution consists of isolated configurations, far away from each other [24, 25]. Apart from the XORSAT problem [26, 27, 28] which can be solved by Gaussian elimination, the random LOPs are very hard to solve in a broad region of the density of constraints, below their SAT-UNSAT transition. For these LOPs, it is known that SP is equivalent to BP. The BP+decimation method has been found to give rather poor results, and the BP+reinforcement, which works better, is still rather limited. One reason for this hardness is the fact that local marginals often convey little information on the solution.

This situation has motivated us to explore some extensions of the message-passing approaches, in which one estimates, beside local marginals, some correlation functions of the variables. Several possibilities to obtain information on the correlations from message-passing procedures have been explored recently [29, 30, 31, 32, 33, 34]. Here we use the susceptibility propagation initially introduced in [32]. We first study the general convergence properties of this method,

and the reliability of its determination of correlations. We show that some of the hard LOPs that could not be solved by previous methods can now be solved by a mixture of the single-variable decimation with a new pair-decimation procedure which makes use of the knowledge of correlation. In the case of binary variables which we study here, this new procedure amounts to identifying a strongly correlated pair of variables, and fixing the relative orientation of the two variables.

The paper is organised as follows. In Sect. 2, we define the occupation problems. Sect. 3 introduces the message passing procedures, belief propagation and susceptibility propagation, and gives some simple basic properties of susceptibility propagation. Sect. 4 explains the new decimation procedure based on the estimate of correlations between pairs of variables. In Sect. 5, it is applied numerically to locked occupation problems and the accuracy of the method is examined: we measure the performance of the decimation process which makes use of the correlations obtained with this method. A short conclusion is contained in Sect. 6.

2. Occupation Problems

2.1. Definition

We consider a subclass of constraint satisfaction problems, the occupation problems [24, 25]. In these problems the elementary variables are binary, and the value of $x_i \in \{0, 1\}$ is interpreted as an occupation number of a point $i \in \{1, \dots, N\}$. These N occupation variables are related by M constraints $a \in \{1, \dots, M\}$. The constraint a applies to the occupation variables of k_a points, $\partial a = \{i_{a,1}, \dots, i_{a,k_a}\}$, forming a subset of size k_a of $\{1, \dots, N\}$. The constraint acts on the number of occupied variables in this subset, $r_a = \sum_{i \in \partial a} x_i$. It is parametrized by a $(k+1)$ -component ‘‘constraint-vector’’ $A_a = (A_a(0), \dots, A_a(k_a))$ with binary entries: by definition, the constraint a is satisfied ($\psi_a = 1$) if and only if $A_a(r_a) = 1$.

Many well known CSPs belong to the category of occupation problems. For instance, k -XORSAT [26] is described by alternating constraint vectors $A = (0, 1, 0, 1, 0, 1, \dots)$; 1-in- k satisfiability is described by $A = (0, 1, 0, 0, 0, \dots)$. We shall often denote a problem by the set of allowed values in the occupation-vector. For instance, 1-or-2-or-4 in 5 satisfiability corresponds to $A = (0, 1, 1, 0, 1, 0)$.

A factor graph $G = (V; E)$ can be associated with an instance of an occupation problem [19]. Each variable and each constraint becomes a vertex of this graph (so that $|V| = N + M$), and an edge connects variable i and constraint a if and only if $i \in \partial a$. The neighborhood of the vertices are $\partial a = \{i \in V | (i, a) \in E\}$, $\partial i = \{a \in V | (i, a) \in E\}$. For a collection of variables in $S \subset V$, we shall write $\underline{x}_S = \{x_i | i \in S\}$. We also use the short-hand notation $\underline{x} = \underline{x}_V$. The factor graph representation of a locked occupation problem (1-in-4 satisfiability $A = (0, 1, 0, 0, 0)$) with a satisfying assignment is shown in Fig. 1.

An occupation problem is *locked* if the following three conditions are met [24, 25, 35]

- $A(0) = A(k) = 0$.
- $A(r)A(r+1) = 0$ for $r = 0, \dots, k-1$.
- Each variable appears in at least two constraints: $|\partial i| \geq 2$

In a locked problem, each satisfiable assignment is an isolated point (one cannot obtain another assignment satisfying all constraints by changing a small number of variables). This feature seems to be at the origin of the difficulty to solve these problems with the usual message passing methods [24, 25].

2.2. Random locked occupation problems

We shall study random instances of locked occupation problems defined as follows: All the function nodes have degree $k_a = K$, and the constraint-vectors are all equal to the same vector A . The factor graph is uniformly chosen among the graphs where all function nodes have degree

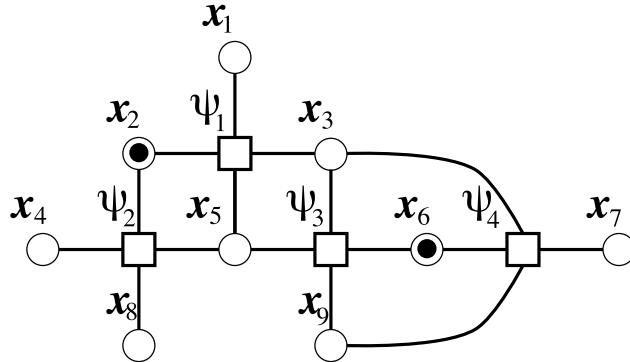


Figure 1. The factor graph representation of a small instance of 1-in-4 satisfiability ($A = (0, 1, 0, 0, 0)$). It has $N = 9$ variables (circles) and $M = 4$ constraints (squares) $\psi_1(x_1, x_2, x_3, x_5)$, $\psi_2(x_2, x_4, x_5, x_8)$, $\psi_3(x_3, x_5, x_6, x_9)$ and $\psi_4(x_3, x_6, x_7, x_9)$. This figure also shows a satisfying assignment in which x_2 and x_6 are occupied while other sites are not.

K and the variables have random degrees which are independent identically distributed variables chosen from the truncated Poisson degree distribution

$$q(\ell) = \begin{cases} 0 & (\ell = 0, 1) \\ \frac{e^{-c} c^\ell}{\ell! (1 - (1+c)e^{-c})} & (\ell \geq 2) \end{cases}. \quad (1)$$

This ensures that all variable nodes have degree at least 2, making the problem locked; the average degree of a variable is

$$\bar{\ell} = \sum_{\ell=0}^{\infty} \ell q(\ell) = \frac{c(1 - e^{-c})}{(1 - (1+c)e^{-c})}. \quad (2)$$

and the average number of variable nodes, \bar{N} , is given by $\bar{N} = KM/\bar{\ell}$.

The ‘thermodynamic limit’ is taken by sending $M \rightarrow \infty$ at fixed $\bar{\ell}$, so that the density of constraints M/\bar{N} is fixed. The phase diagram has been studied in [24]. When increasing $\bar{\ell}$, the probability that a satisfying assignment exists drops from 1 to 0 at the ‘satisfiability threshold’ ℓ_s . Between the ‘clustering threshold’ ℓ_d ([24] and the satisfiability threshold, the system is in a SAT phase, but the solutions are isolated points which are far away from each other. In this regime, although the satisfying assignments still exist with probability one, it is very difficult to find one by the algorithms known so far, because of the splitting of the set of solutions into clusters. This is the region which we will focus on most.

3. Message passing methods

3.1. Belief Propagation Update Rules

Consider an occupation problem described by a factor graph $G = (V, E)$ and a constraint-vector A . The uniform measure over all satisfying assignments (assuming that there is at least one such assignment), is defined as

$$p(\underline{x}) = \frac{1}{Z} \prod_{a=1}^M \psi_a(\underline{x}_{\partial a}) \quad (3)$$

where the function $\psi_a(\underline{x}_{\partial a})$ is equal to 1 if $A(\sum_{i \in \partial a} x_i) = 1$, and to 0 otherwise.

For later use, we introduce local ‘external fields’ h_ℓ^x ($x \in \{0, 1\}$, $\ell \in V$), which will be sent to zero at the end, and consider a joint probability distribution

$$p(\underline{x}|\underline{h}^x) = \frac{1}{Z(\underline{h}^x)} \prod_{a=1}^M \psi_a(\underline{x}_{\partial a}) \times \prod_{\ell=1}^N \prod_x e^{h_\ell^x \delta_{x_\ell, x}}. \quad (4)$$

This probability distribution is well defined as soon as there exists at least one (“SAT”) configuration satisfying all the constraints. The constant $Z(\underline{h}^x)$ is a normalization factor. Our final aim is to extract solutions from the uniform measure $p(\underline{x}|0)$ over solutions satisfying all constraints (when there exists at least one solution).

The marginal distribution $p_i(x_i|\underline{h})$ can be estimated by the BP algorithm. The BP update rules for two families of messages, namely cavity fields and cavity biases, are given by [36, 1]

$$\nu_{i \rightarrow a}^{(t+1)}(x_i|\underline{h}^x) = \frac{1}{Z_{i \rightarrow a}^{(t)}(\underline{h}^x)} \prod_{b \in \partial i \setminus a} \hat{\nu}_{b \rightarrow i}^{(t)}(x_i|\underline{h}^x) \times \prod_x e^{h_i^x \delta_{x_i, x}}, \quad (5)$$

$$\hat{\nu}_{a \rightarrow i}^{(t)}(x_i|\underline{h}^x) = \sum_{\underline{x}'_{\partial a}} \delta_{x_i, x'_i} \psi_a(\underline{x}'_{\partial a}) \prod_{\ell \in \partial a \setminus i} \nu_{\ell \rightarrow a}^{(t)}(x'_\ell|\underline{h}^x). \quad (6)$$

Here, we have decided to introduce a normalization factor $Z_{i \rightarrow a}^{(t)}(\underline{h}^x)$ for $\nu_{i \rightarrow a}^{(t)}(x_i|\underline{h}^x)$ and to avoid the normalization for $\hat{\nu}_{a \rightarrow i}^{(t)}(x_i|\underline{h}^x)$. This choice is perfectly valid for BP, and it helps to get relatively simple susceptibility propagation update rules (10)(11).

Assuming convergence to a fixed point, the BP estimate for the marginal distribution of variable i is:

$$p_i(x_i|\underline{h}^x) = \frac{1}{Z_i(\underline{h}^x)} \prod_{b \in \partial i} \hat{\nu}_{b \rightarrow i}^{(*)}(x_i|\underline{h}^x), \quad (7)$$

where $\hat{\nu}_{a \rightarrow i}^{(*)}(x_i|\underline{h}^x)$ is the fixed point of the BP iteration.

3.2. Susceptibility Propagation Update Rules

The 2-point connected correlation function at $\underline{h} = 0$ is obtained as

$$p_{ij}^{\text{conn}}(x_i, x_j) \equiv p_{ij}(x_i, x_j) - p_i(x_i)p_j(x_j) = \left. \frac{\partial p_i(x_i|\underline{h}^x)}{\partial h_j^{x_j}} \right|_{\underline{h}=0}. \quad (8)$$

To have a message-passing algorithm to calculate this quantity, we introduce the cavity susceptibility and its companion by

$$\nu_{i \rightarrow a, j}(x_i, x_j) = \left. \frac{\partial \nu_{i \rightarrow a}(x_i|\underline{h}^x)}{\partial h_j^{x_j}} \right|_{\underline{h}=0}, \quad ; \quad \hat{\nu}_{a \rightarrow i, j}(x_i, x_j) = \left. \frac{\partial \hat{\nu}_{a \rightarrow i}(x_i|\underline{h}^x)}{\partial h_j^{x_j}} \right|_{\underline{h}=0}. \quad (9)$$

Note that the roles of variables x_i and x_j are asymmetric; j can be an arbitrary variable while i is a neighbor of the constraint a .

The cavity susceptibility can be calculated by a message-passing method [29]. The susceptibility propagation update rules can be obtained by differentiating the belief propagation update rules (5) and (6) with respect to h_j^x . They read [32, 37]

$$\nu_{i \rightarrow a, j}^{(t+1)}(x_i, x_j) = \frac{1}{Z_{i \rightarrow a}^{(t)}(\underline{h}^x)} \prod_{b \in \partial i \setminus a} \left(\delta_{i,j} \delta_{x_i, x_j} + \sum_{b \in \partial i \setminus a} \frac{\hat{\nu}_{b \rightarrow i, j}^{(t)}(x_i, x_j)}{\hat{\nu}_{b \rightarrow i}^{(t)}(x_i)} + C_{i \rightarrow a, j}^{(t)}(x_j) \right), \quad (10)$$

$$\hat{\nu}_{a \rightarrow i, j}^{(t)}(x_i, x_j) = \sum_{\underline{x}'_{\partial a}} \delta_{x_i, x'_i} \psi_a(\underline{x}'_{\partial a}) \times \left(\prod_{\ell \in \partial a \setminus i} \nu_{\ell \rightarrow a}^{(t)}(x'_\ell) \right) \sum_{m \in \partial a \setminus i} \frac{\nu_{m \rightarrow a, j}^{(t)}(x'_m, x_j)}{\nu_{m \rightarrow a}^{(t)}(x'_m)}, \quad (11)$$

where

$$\nu_{i \rightarrow a}^{(t)}(x_i) = \nu_{i \rightarrow a}^{(t)}(x_i | \underline{h}^x = \underline{0}), \quad \hat{\nu}_{a \rightarrow i}^{(t)}(x_i) = \hat{\nu}_{a \rightarrow i}^{(t)}(x_i | \underline{h}^x = \underline{0}). \quad (12)$$

The function $C_{i \rightarrow a,j}^{(t)}(x_j)$ originates from the derivative of $Z_{i \rightarrow a}^{(t)}$ and can be determined by requiring the normalization $\sum_{x_i} \nu_{i \rightarrow a,j}^{(t)}(x_i, x_j) = 0$.

Let us suppose that we have found a fixed point $(\nu_{i \rightarrow a}^{(*)}, \hat{\nu}_{a \rightarrow i}^{(*)}, \nu_{i \rightarrow a,j}^{(*)}, \hat{\nu}_{a \rightarrow i,j}^{(*)})$ of BP and the susceptibility propagation. By differentiating (7) with respect to the external fields, we can express the 2-point connected correlation function in terms of the messages at the fixed point as

$$p_{ij}^{\text{conn}}(x_i, x_j) = p_i(x_i)[\delta_{i,j}\delta_{x_i,x_j} + C_{ij}(x_j)] + \frac{1}{Z_i(\underline{0})} \sum_{b \in \partial i} \hat{\nu}_{b \rightarrow i,j}^{(*)}(x_i, x_j) \prod_{c \in \partial i \setminus b} \hat{\nu}_{c \rightarrow i}^{(*)}(x_i). \quad (13)$$

The constant $C_{ij}(x_j)$ is related to the derivative of $Z_i(\underline{h})$ and is conveniently fixed by the condition $\sum_{x_j} p_{ij}^{\text{conn}}(x_i, x_j) = 0$.

3.3. Log-likelihood representation

The rules (10,11) apply to all types of CSPs with discrete variables. When dealing with binary variables, it is helpful to rewrite the belief and susceptibility update equations in terms of log-likelihood variables. We introduce the cavity field and cavity bias in the log-likelihood representation $n_{i \rightarrow a}$ and $\hat{n}_{a \rightarrow i}$ as (we omit the time superscript (t) where it is obvious):

$$\nu_{i \rightarrow a}(x_i | \underline{h}) = A_{i \rightarrow a} e^{n_{i \rightarrow a}(\underline{h})s_i}, \quad ; \quad \hat{\nu}_{a \rightarrow i}(x_i | \underline{h}) = B_{a \rightarrow i} e^{\hat{n}_{a \rightarrow i}(\underline{h})s_i}, \quad (14)$$

where s_i is the spin variable $s_i = 2x_i - 1 = \pm 1$ and the external fields in the two representations are related by $h_j = \frac{h_j^1 - h_j^0}{2}$.

Naturally we define the cavity susceptibility in the log-likelihood representation as

$$\eta_{i \rightarrow a,j} = \left. \frac{\partial n_{i \rightarrow a}(\underline{h})}{\partial h_j} \right|_{\underline{h}=0}; \quad \hat{\eta}_{a \rightarrow i,j} = \left. \frac{\partial \hat{n}_{a \rightarrow i}(\underline{h})}{\partial h_j} \right|_{\underline{h}=0}. \quad (15)$$

The belief propagation update rules read

$$n_{i \rightarrow a}^{(t+1)} = \sum_{b \in \partial i \setminus a} \hat{n}_{b \rightarrow i}^{(t)} + h_i; \quad \hat{n}_{a \rightarrow i}^{(t)} = f_{a \rightarrow i}(\{n_{j \rightarrow a}^{(t)}\}_{j \in \partial a \setminus i}), \quad (16)$$

where

$$f_{a \rightarrow i}(\{n_{j \rightarrow a}\}_{j \in \partial a \setminus i}) = \frac{1}{2} \log \frac{F(+1)}{F(-1)}; \quad F(\sigma) = \sum_{\underline{s}_{\partial a}} \delta_{s_i, \sigma} \psi_a(\underline{s}_{\partial a}) \prod_{j \in \partial a \setminus i} e^{n_{j \rightarrow a} s_j}. \quad (17)$$

By differentiating both sides of (16), we obtain

$$\eta_{i \rightarrow a,j}^{(t+1)} = \sum_{b \in \partial i \setminus a} \hat{\eta}_{b \rightarrow i,j}^{(t)} + \delta_{i,j} \quad (18)$$

$$\hat{\eta}_{a \rightarrow i,j}^{(t)} = \sum_{m \in \partial a \setminus i} \frac{\partial f_{a \rightarrow i}(\{n_{j \rightarrow a}^{(t)}\}_{j \in \partial a \setminus i})}{\partial n_{m \rightarrow a}} \times \eta_{m \rightarrow a,j}^{(t)}. \quad (19)$$

Assuming that a solution $n_{j \rightarrow a}^{(t)}$ of the BP equations (16) is used, one sees that the susceptibility propagation update rule (18,19) is an inhomogeneous linear system in η and $\hat{\eta}$. The coefficient matrix takes the following form:

$$\frac{\partial f_{a \rightarrow i}(\{n_{j \rightarrow a}\}_{j \in \partial a \setminus i})}{\partial n_{m \rightarrow a}} = \frac{\langle s_m s_i \rangle - \langle s_m \rangle \langle s_i \rangle}{1 - \langle s_i \rangle^2}. \quad (20)$$

Here the brackets in $\langle s_i \rangle$ and $\langle s_m s_i \rangle$ for $i, m \in \partial a$ mean the expectation value with respect to the joint probability distribution for variables that are neighbors of a constraint; it can be obtained solely from beliefs [1, Sec.14.2.3].

The magnetization and the pair correlation between arbitrary sites are given in terms of the fixed-point messages by

$$\langle s_i \rangle = \tanh \left(\sum_{b \in \partial i} \hat{n}_{b \rightarrow i}^{(*)} \right), \quad (21)$$

$$\langle s_i s_j \rangle_{\text{conn}} \equiv \langle s_i s_j \rangle - \langle s_i \rangle \langle s_j \rangle = \left[1 - \tanh^2 \left(\sum_{b \in \partial i} \hat{n}_{b \rightarrow i}^{(*)} \right) \right] \times \left[\sum_{c \in \partial i} \hat{n}_{c \rightarrow i, j}^{(*)} + \delta_{i,j} \right]. \quad (22)$$

Although the symmetry between the sites is not manifest in the right-hand side of (22), it is in fact symmetric when the graph is a tree and the fixed point gives the exact magnetization.

3.4. Basic properties of susceptibility propagation

In order to study the structure of susceptibility propagation update rules (18,19), we construct a kMN -component column vector

$$\mathbf{y}^{(t)} = (\eta_{i \rightarrow a, j}^{(t)}, \hat{\eta}_{a \rightarrow i, j}^{(t)})_{(i,a) \in E, j \in V}^t. \quad (23)$$

Then the fixed point condition associated with (18,19) can be written as a linear equation

$$\mathbf{y}^{(*)} = \mathbf{M} \mathbf{y}^{(*)} + \mathbf{b}, \quad (24)$$

with the inhomogeneous term

$$\mathbf{b} = (\delta_{i,j}, 0)_{(i,a) \in E, j \in V}^t. \quad (25)$$

The coefficient matrix is block-diagonal in j :

$$\mathbf{M}_{(iaj), (i' a' j')} = \delta_{j,j'} \mathcal{M}_{(ia), (i' a')}, \quad (26)$$

$$\mathcal{M} = \begin{pmatrix} 0 & \mathbf{1}(a' \in \partial i \setminus a) \delta_{i,i'} \\ \mathbf{1}(i' \in \partial a \setminus i) \delta_{a,a'} \frac{\partial f_{a \rightarrow i}(\{n_{j \rightarrow a}^{(*)}\}_{j \in \partial a \setminus i})}{\partial n_{i' \rightarrow a'}} & 0 \end{pmatrix}, \quad (27)$$

where the block \mathcal{M} is independent of the block index j .

Thus we obtain the unique fixed point

$$\mathbf{y}^{(*)} = (\mathbf{1} - \mathbf{M})^{-1} \mathbf{b} \quad (28)$$

if $(\mathbf{1} - \mathbf{M})$ is invertible, or equivalently, if $(\mathbf{1} - \mathcal{M})$ is invertible.

The susceptibility propagation update rules (18,19) can be regarded as an iterative method to solve the linear equation (28). It converges to a value irrespective of the initial vector if all the eigenvalues of \mathcal{M} have moduli smaller than unity. Because the block \mathbf{M} does not depend on

j , the existence of the fixed points and convergence to them are solely determined by \mathcal{M} and do not depend on j .

When the factor graph is a tree, even in presence of the external fields \underline{h}^x , the exact marginals are obtained by (7) on a fixed point $\nu_{i \rightarrow a}^{(*)}, \hat{\nu}_{i \rightarrow a}^{(*)}$ [19]. Therefore, by differentiation with respect to \underline{h}^x , there exists a susceptibility fixed-point which gives the exact 2-point correlation function. In the examples which we have considered, the iteration of susceptibility propagation converges to this fixed-point. On the other hand, if the graph has more than one loop, there is no guarantee either that the fixed point exists or the iteration leads to that fixed point.

A simple test of these statements is obtained by studying the 1-in-2 satisfiability problem, which is nothing but the anti-ferromagnetic Ising model at zero temperature, or a version of the XORSAT problem [26]. If one considers this problem on a chain of length N , it is easy to check that the belief and susceptibility propagation equations have a unique fixed point, which gives the correct results for the magnetization and for the pair correlation ($\langle s_i \rangle = 0$ and $\langle s_i s_j \rangle - \langle s_i \rangle \langle s_j \rangle = (-1)^{i-j}$).

Consider now the same problem on the simplest graph with one loop, a ring. The BP iteration now has a continuous family of fixed points, characterized by ‘right moving’ messages which alternate between the value A_+ and $-A_+$, and ‘left moving’ messages which alternate between the value A_- and $-A_-$, where A_\pm are two constants [1]. As a consequence of the existence of this family of fixed points, $(1 - \mathcal{M})$ is not invertible; in fact it has an eigenvector with zero eigenvalue, $\mathbf{y}_0 = (\mathbf{1}, -\mathbf{1})$ where $\mathbf{1}$ corresponds to the η -block and $-\mathbf{1}$ corresponds to the $\hat{\eta}$ block. In agreement with the existence of this dangerous eigenvector, one finds that the susceptibility propagation update rule does not converge. As the susceptibility messages are updated, $\eta_{i \rightarrow i+\frac{1}{2},j}$ picks up the constant shift $\delta_{i,j} = 1$. This effect is accumulated as the messages go around the ring, and the consequence is that the messages diverge as $t \rightarrow \infty$.

In summary, for 1-in-2 satisfiability on a ring, the belief propagation can converge to a family of solutions for the magnetization among which only one solution is exact. On the other hand, the susceptibility propagation update does not have a fixed point, it diverges. In the simple case of a ring, this behaviour can be cured by using the finite temperature version of the BP and susceptibility propagation update equations. But in general there is no guarantee of convergence of loopy BP and loopy susceptibility propagation, and when they converge the quality of their results cannot be assessed a priori. Fig. 2 gives an example of analysis of a small instance of 1-in-4 satisfiability, giving an idea of the errors made by susceptibility propagation on small factor graphs. On the other hand, as for standard BP, one may hope that the method becomes better for large instances when the factor graph is locally tree-like.

4. Correlation based decimation

4.1. Single-variable and pair decimation

As we mentioned in the introduction, decimation consists in finding a strongly polarized variable and fixing it to its most probable value. One way to measure the degree of polarization is to compute the marginal $p_i(x)$ of a variable x_i , and the corresponding entropy $S_i = -\sum_x p_i(x) \log p_i(x)$. At each step one chooses the variable with the smallest entropy and fixes it: one gets a new occupation problem which has one variable less, and one iterates the procedure.

Assuming that the susceptibility propagation provides us with the good estimate for the 2-point connected correlation, we can think of a decimation procedure which acts on a pair of variables instead of a single variable. Let x_i and x_j be variables. If one defines a random variable $y_{ij} = \mathbf{1}(x_i = x_j)$, one can compute the probability $p_{ij}(y_{ij})$ of y_{ij} , and the corresponding pair entropy $S_{ij} = -\sum_y p_{ij}(y) \log p_{ij}(y)$, once one knows the 2-variable marginal $p_{ij}(x_i, x_j) = p_i(x_i)p_j(x_j) + p_{ij}^{\text{conn}}(x_i, x_j)$. In pair decimation, one identifies the pair (i, j) with the smallest entropy S_{ij} , and one fixes the “pair relation” as either $x_i = x_j$ or $x_i + x_j = 1$,

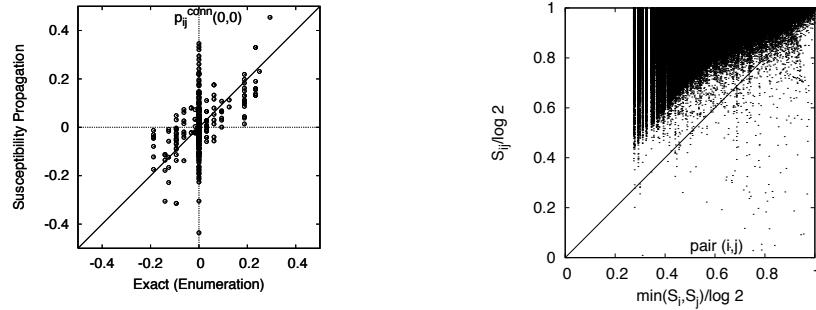


Figure 2. Left: Comparison between the 2-point connected correlation function calculated exactly and that estimated with susceptibility propagation. A 1-in-4 satisfiability instance on a randomly generated factor graph with $N = 27$ variables and $M = 16$ constraints with Poisson degree distribution with average degree $\bar{l} = 2.4856$. Right: Comparison between the minimum entropy $\min(S_i, S_j)$ (where S_i and S_j are the entropies of x_i, x_j) and S_{ij} , that of $y_{ij} = \mathbf{1}(x_i = x_j)$. For each pair i, j , we plot $S_{ij}/\log 2$ versus $\min(S_i, S_j)/\log 2$. The points in the bottom-right half part correspond to correlated pairs. The instance is 1-in-4 satisfiability on a random factor graph with $N = 1618$ variables and $M = 1000$ factors with the truncated Poisson degree distribution with average degree $\bar{l} = 2.4856$.

depending on which event is the most probable according to the measured correlation. This results in a reduced smaller CSP, which is still an occupation problem.

In both cases, it is convenient, when one fixes a variable (or the relation between two variables), to find out if this has direct logical implications on other variables. These implications, called “unit clause propagation” in the context of satisfiability, can be found efficiently by the “warning propagation” algorithm, a min-sum message passing procedure described in [28, 22].

The efficiency of the new pair decimation process depends on whether one can find a pair with less entropy than the single variable with the smallest entropy. It is easy to see that, in the absence of correlations, namely if $p_{ij}(x_i, x_j) = p_i(x_i)p_j(x_j)$, then the entropy of y_{ij} is larger than the one of x_i or x_j . So the whole procedure relies on being able to detect correlations. Fig. 2 shows that strongly correlated pairs can be found, making it possible to use pair decimation efficiently.

4.2. Algorithm

Our algorithm has two main parts, the message passing (belief + susceptibility propagation) part and the decimation part. The basic message-passing algorithm that we use is described by the following pseudocode:

Input: Factor graph, constraint-vector, convergence criterion, initial messages
Output: Estimate for 2-point connected correlation functions (or ERROR-NOT-CONVERGED)

- Initialize messages
- Repeat until all messages have converged to a fixed point (within a given resolution):
 - Update cavity fields and cavity biases $\nu_{i \rightarrow a}^{(t)}(x_i)$ and $\hat{\nu}_{a \rightarrow i}^{(t)}(x_i)$ with (5)
 - Update cavity susceptibilities $\nu_{i \rightarrow a,j}^{(t)}(x_i, x_j)$ and $\hat{\nu}_{a \rightarrow i,j}^{(t)}(x_i, x_j)$ with (10)(11) with the help of $\nu_{i \rightarrow a}^{(t)}(x_i)$ and $\hat{\nu}_{a \rightarrow i}^{(t)}(x_i)$ obtained above
- Compute 1-variable marginals $p_i(x_i)$ and entropies S_i from the fixed-point messages $\hat{\nu}_{a \rightarrow i}^{(*)}(x_i)$ by (7)

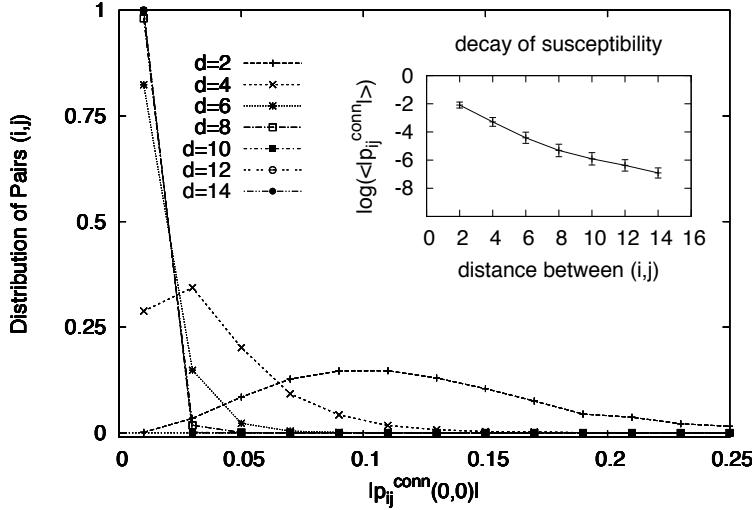


Figure 3. This graph shows how the 2-point connected correlation $p_{ij}^{\text{conn}}(x_i, x_j)$ decays as the distance d between x_i and x_j increases. At each distance d , the distribution of $|p_{ij}^{\text{conn}}(0,0)|$ is plotted. In the inset, logarithm of the average of that quantity is plotted against the distance. The instance is 1-in-4 satisfiability on a random factor graph with $N = 1618$ variables and $M = 1000$ factors with the truncated Poisson degree distribution with average degree $\bar{\ell} = 2.4856$.

- Compute 2-point connected correlation functions $p_{ij}^{\text{conn}}(x_i, x_j)$ and entropies S_{ij} from the fixed-point messages $\hat{\nu}_{a \rightarrow i}^{(*)}(x_i)$ and $\hat{\nu}_{a \rightarrow i,a}^{(*)}(x_i, x_j)$ by (13)

This algorithm requires a memory proportional to kMN , and each step of iteration requires a computation of $\mathcal{O}(N^2)$ for fixed k .

One accelerates it by observing that the correlations tend to decay rather fast with the distance between variables on the factor graph. Fig. 3 shows the distribution of magnitude of 2-point connected correlation function computed with susceptibility propagation for all pairs of points in a graph for a fixed distance between the points. One observes a broad dispersion of correlations, and an approximate exponential decay with the distance. Here we measure the distance d with the convention that each edge connecting a variable to a constraint is of length 1. Because of this exponential decay, it is possible to use in some cases approximate versions of susceptibility propagation which are faster and use less memory. This is done by truncating to zero the cavity susceptibilities $\nu_{i \rightarrow a,j}, \hat{\nu}_{a \rightarrow i,j}$ beyond some prescribed distance $\text{dist}(a,j) > d$ or $\text{dist}(i,j) > d$ and keeping only the correlation functions between pairs of variables not far from each other. This truncation provides us with an efficient practical method to compute the 2-variable correlations between variables with $d \geq 4$.

The decimation algorithm that we have used is described by the following pseudo-code:

Input: Factor graph, constraint-vector, convergence criterion, initial messages, threshold entropy S_{th} .

Output: A satisfying assignment (or FAIL-NOT-FOUND)

- While graph has more than R variables:
 - Run the message passing algorithm (belief + susceptibility propagation)
 - Find the variable i with the smallest entropy S_i
 - Find the pair ij with the smallest entropy S_{ij}
 - if $(S_i < S_{\text{th}} \text{ or } S_i < S_{ij})$, fix variable i

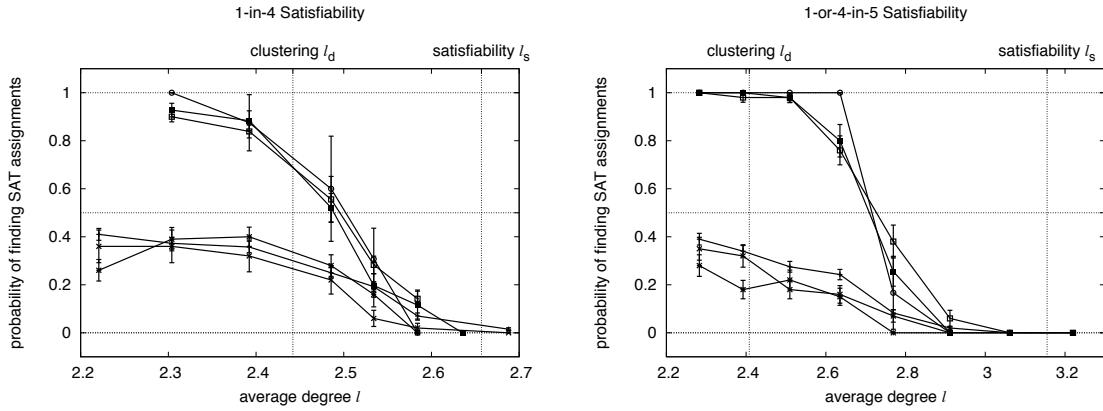


Figure 4. Success probability of pair decimation process for 1-in-4 satisfiability $A = (0, 1, 0, 0, 0)$ (left) and 1-or-4-in-5 satisfiability $A = (0, 1, 0, 0, 1, 0)$ (right) on a random factor graph with M constraints and average degree \bar{l} (right), plotted versus \bar{l} . For comparison, the performance of simple belief-guided decimation process is shown. The vertical lines show the clustering and satisfiability thresholds.

- else, fix the pair relation ij
- Clean the graph :
 - * Fix the value of isolated variables
 - * Do warning propagation
 - * Find degree 2 constraints, and if they enforce that $y_{ij} = 0$ or that $y_{ij} = 1$, fix the pair relation
- When the number of variables is equal to or smaller than R : perform an exhaustive search for satisfying assignments. If found
 - Then return the satisfying assignment
 - Else return FAIL-NOT-FOUND

The threshold entropy $S_{\text{th}} = 0$ is an ad-hoc parameter. For the optimal reduction of the entropy within a decimation step, it is reasonable to set $S_{\text{th}} = 0$. However, we have found that $S_{\text{th}} > 0$ performs better for finding a satisfying assignment. The optimal value of S_{th} depends on the type of locked occupation model and the average degree. This behaviour can be understood from the fact that the estimation of 1-variable marginals is more precise than the 2-variable ones within given computational resource. Thus it is advantageous to respect the former if it is decisively small.

5. Numerical results

We have run some simulations of random LOPs using the above algorithm.

The convergence criterion for message passing was set to 10^{-2} . The threshold for exhaustive search has been fixed to $R = 16$, and the values of S_{th} were typically 0.20. The performance of this algorithm is shown for 1-in-4 satisfiability $A = (0, 1, 0, 0, 0)$ and 1-or-4-in-5 satisfiability $A = (0, 1, 0, 0, 1, 0)$ in Fig. 4. For 1-in-4 satisfiability, data with randomization is presented: instead of fixing the most polarized variable or pair, we fix a variable or pair randomly chosen among a fixed number (here we adopt 8) of most polarized variables/pairs. The figure also shows the two important thresholds for these problems, ℓ_d and ℓ_s , which are values of the average degree (a measure of the density of constraints) separating qualitatively distinct phases. In both LOPs the performance of the new algorithm using pair decimation is clearly improved compared to the simple belief-guided decimation employed in [24]. Especially for 1-or-4-in-5,

the present algorithm works well above the clustering threshold, a region of $\bar{\ell}$ where all known algorithms are reported to perform poorly [24].

6. Conclusion and Discussion

We have shown how to find satisfying assignments for locked occupation problems in their hard (clustered) phase by using a new pair decimation technique based on the measurement of correlation among variables. This improves significantly upon the conventional decimation method which is guided by 1-variable marginals only. The intuitive understanding of this behaviour is that, since flipping a variable in an LOP typically forces another variable far away to be flipped, the performance of the algorithm is improved when one estimates the correlations.

Acknowledgment

S.H. was supported by Ryukoku University Research Fellowship (2008).

References

- [1] Mézard M and Montanari A 2009 *Information, Physics and Computation* (New York: Oxford University Press)
- [2] Cheeseman P, Kanefsky B and Taylor W M 1991 *Proc. 12th Int. Joint Conf. on Artificial Intelligence* ed J M and R R (San Mateo, CA: Morgan Kaufmann) pp 331–7
- [3] Mitchell D G, Selman B and Levesque H J 1992 *Proc. 10th Nat. Conf. on Artificial Intelligence* (Menlo Park, CA: AAAI Press) pp 459–65
- [4] Kirkpatrick S and Selman B 1994 *Science* **264** 1297–301
- [5] Selman B, Mitchell D G and Levesque H J 1996 *Artificial Intelligence* **81** 17–29
- [6] Mézard M, Parisi G and Zecchina R 2002 *Science* **297** 812–5
- [7] Mézard M and Zecchina R 2002 *Phys. Rev. E* **66** 56126
- [8] Mertens S, Mézard M and Zecchina R 2006 *Random Structure and Algorithms* **28** 340–73
- [9] Mulet R, Pagnani A, Weigt M and Zecchina R 2002 *Phys. Rev. Lett.* **89** 268701
- [10] Krzakala F, Montanari A, Ricci-Tersenghi F, Semerjian G and Zdeborová L 2007 *Proc. of the National Academy of Sciences* **104** 10318–23
- [11] Zdeborová L and Krzakala F 2007 *Phys. Rev. E* **76** 031131
- [12] Biroli G, Monasson R and Weigt M 2000 *Eur. Phys. J. B* **14** 551–68
- [13] Friedgut E 1999 *J. Amer. Math. Soc.* **12** 1017–54
- [14] Mézard M, Mora T and Zecchina R 2005 *Phys. Rev. Lett.* **94** 197205
- [15] Achlioptas D and Ricci-Tersenghi F 2006 *Proc. of 38th annual ACM symposium on Theory of Computing* (New York: ACM) pp 130–9
- [16] Bethe H A 1935 *Proc. Roy. Soc. London A* **150** 552–75
- [17] Gallager R 1962 *IRE Transactions on Information Theory* **8** 21–8
- [18] Pearl J 1982 *Proc. of American Association of Artificial Intelligence National Conference on AI* (Pittsburgh: AAAI Press) pp 133–6
- [19] Kschischang F, Frey B and Loeliger H 2001 *IEEE Transactions on Information Theory* **47** 498–519
- [20] Chavas J, Furtlechner C, Mézard M and Zecchina R 2005 *J. Stat. Mech.* P11016
- [21] Braunstein A and Zecchina R 2006 *Phys. Rev. Lett.* **96** 030201
- [22] Montanari A, Ricci-Tersenghi F and Semerjian G 2007 Preprint arXiv:0709.1667
- [23] Ricci-Tersenghi F and Semerjian G 2009 *J. Stat. Mech.* **9** P09001
- [24] Zdeborová L and Mézard M 2008 *Phys. Rev. Lett.* **101** 078702
- [25] Zdeborová L and Mézard M 2008 *J. Stat. Mech.* P12004
- [26] Creignou N and Daude H 1999 *Discrete Applied Mathematics* **96** 41–53
- [27] Cocco S, Dubois O, Mandier J and Monasson R 2003 *Phys. Rev. Lett.* **90** 047205
- [28] Mézard M, Ricci-Tersenghi F and Zecchina R 2003 *Journal of Statistical Physics* **111** 505–33
- [29] Montanari A and Rizzo T 2005 *J. Stat. Mech.* **10** P10011
- [30] Chertkov M and Chernyak V 2006 *Phys. Rev. E* **73** 065102
- [31] Parisi G and Slanina F 2006 *J. Stat. Mech.* L02003
- [32] Mézard M and Mora T 2009 *J. of Physiology-Paris* **103** 107–13 preprint arXiv:0803.3061
- [33] Tanaka K 2003 *IEICE Transactions on Information and Systems* **E86-D** 1228–42
- [34] Welling M and Teh Y W 2004 *Neural Computation* **16** 197–211

- [35] Zdeborová L 2008 *Statistical Physics of Hard Optimization Problems* Ph.D. thesis Université Paris-Sud preprint arxiv:0806.4112
- [36] Yedidia J, Freeman W and Weiss Y 2003 *Understanding belief propagation and its generalizations* (San Francisco: Morgan Kaufmann) pp 239–236
- [37] Mora T 2007 *Géométrie et inférence dans l'optimisation et en théorie de l'information* Ph.D. thesis Université Paris-Sud <http://tel.archives-ouvertes.fr/tel-00175221/en/>