Journal of Statistical Mechanics:
Theory and Experiment

Palette-colouring: a belief propagation approach

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

Palette-colouring: a belief propagation approach

Alessandro Pelizzola[1,2,3], **Marco Pretti**[4,6] **and Jort van Mourik**[5]

[1] Dipartimento di Fisica, CNISM and Center for Computational Studies, Politecnico di Torino, Corso Duca degli Abruzzi 24, I-10129 Torino, Italy
[2] INFN, Sezione di Torino, Via Pietro Giuria 1, I-10125 Torino, Italy
[3] HuGeF Torino, Via Nizza 52, I-10126 Torino, Italy
[4] CNR—Consiglio Nazionale delle Ricerche, Istituto dei Sistemi Complessi, and CNISM, Dipartimento di Fisica, Politecnico di Torino, Corso Duca degli Abruzzi 24, I-10129 Torino, Italy
[5] Non-Linearity and Complexity Research Group, Aston University, Birmingham B4 7ET, UK
E-mail: alessandro.pelizzola@polito.it, marco.pretti@polito.it and vanmourj@aston.ac.uk

Received 25 February 2011
Accepted 13 April 2011
Published 6 May 2011

Online at stacks.iop.org/JSTAT/2011/P05010
doi:10.1088/1742-5468/2011/05/P05010

Abstract. We consider a variation of the prototype combinatorial optimization problem known as graph colouring. Our optimization goal is to colour the vertices of a graph with a fixed number of colours, in a way to maximize the number of different colours present in the set of nearest neighbours of each given vertex. This problem, which we pictorially call palette-colouring, has been recently addressed as a basic example of a problem arising in the context of distributed data storage. Even though it has not been proved to be NP-complete, random search algorithms find the problem hard to solve. Heuristics based on a naive belief propagation algorithm are observed to work quite well in certain conditions. In this paper, we build upon the mentioned result, working out the correct belief propagation algorithm, which needs to take into account the many-body nature of the constraints present in this problem. This method improves the naive belief propagation approach at the cost of increased computational effort. We also investigate the emergence of a satisfiable-to-unsatisfiable 'phase transition' as a

[6] Author to whom any correspondence should be addressed.

function of the vertex mean degree, for different ensembles of sparse random graphs in the large size ('thermodynamic') limit.

Keywords: cavity and replica method, message-passing algorithms, optimization over networks

Contents

1. Introduction

Graph colouring is a prototype of combinatorial optimization or constraint satisfaction problems [1]. It is NP-complete, so that it can be taken as a benchmark for optimization algorithms. Moreover, it is at the core of a large number of technologically relevant combinatorial problems, such as scheduling. The goal is to assign a colour to each vertex of a given graph (with a fixed number of available colours) in such a way that no pair of vertices connected by an edge have the same colour. Alternatively, one may be satisfied with a suboptimal solution, i.e. minimizing the number of vertex pairs with the same colour.

A nice variant of the above problem has been recently proposed and investigated by Bounkong and co-workers [2, 3]. The variation consists in requiring that the set of colours assigned to each given vertex and its neighbours includes all available colours. The latter problem, which we pictorially call palette-colouring, has been suggested as a basic example of a constraint satisfaction problem arising in the context of distributed data storage [2]. The basic idea is as follows. On a computer network with limited storage resources at each node, it may be convenient to divide a file into a number of segments (colours), which are then distributed over different nodes. Each given node should be able to retrieve the different segments by accessing only its own and nearest-neighbour storage devices, whence the above described constraints. Even in this case, one might be satisfied with a suboptimal solution, i.e. maximizing the number of colours present in each node neighbourhood. We note that palette-colouring has not been proved to be NP-complete, but there is numerical evidence that it becomes intractable for large system sizes [2]. With respect to ordinary colouring, the most relevant difference is that the modified problem becomes easier to solve for graphs with higher, rather than lower, vertex degrees.

In the last few years, different types of constraint satisfaction problems have been faced by message-passing techniques, among which belief propagation (BP) [4, 5]. BP has been originally conceived as a dynamic programming algorithm to perform exact statistical inference for Markov random field models defined on graphs without loops (trees) [6, 7]. Subsequently, it has been demonstrated to be relatively good even for loopy graphs. Such a successful behaviour seems to be related to the fact that actually BP is equivalent to determining a minimum of an approximate free energy function (Bethe free energy) for a corresponding thermodynamic system. The Bethe approximation was indeed very well known to physicists [8, 9], but the connection with BP is a relatively recent result [4].

In [2], Bounkong, van Mourik and Saad analyse an algorithm based on BP, comparing its performance with a variant of Walksat [10]. In particular, the BP-based algorithm makes use of beliefs averaged over several iterations, together with a common decimation strategy. It is observed that, while Walksat works definitely better for small graphs (100 vertices), the opposite occurs for larger (1000 vertices) random graphs. This result is somehow related to the nature of BP itself, since large random graphs are known to be tree-like, in the sense that the probability of finite length loops tends to zero as the number of vertices becomes large. Nevertheless, the BP algorithm employed in [2] follows a naive scheme, in which every message provides a contribution to the probability distribution of a single variable, taking into account a given interaction (constraint). Due to the many-variable nature of the constraints present in the palette-colouring problem, this scheme no longer provides the exact solution even in the case of trees. As suggested in the cited work, the exact solution can be determined at the cost of propagating generalized messages providing a contribution to the joint probability distributions of pairs of nearest-neighbour variables (instead of single-variable distributions). This scheme has already been used in different works dealing with structural and spin glass models (see, for instance, [11] and [12]) in the presence of similar 'all-neighbours' interactions. In the current paper, we work out the pairwise BP scheme for the palette-colouring problem. As readers more familiar with these methods may have noticed, this scheme is not a generalized BP [13]. Indeed, in the literature, the latter term usually denotes a class of algorithms computing the minima of more refined free energy approximations [14], rather than Bethe free energies [15]. Here, however, we derive an algorithm computing the correct Bethe approximation, which is the exact solution for loopless graphs. We then compare the performance of the new BP algorithm (which we shall simply call BP from now on) to the naive one, showing that further improvements can be obtained.

Let us note that the correct Bethe approximation has already been considered for this problem by Wong and Saad [3], in order to investigate the emergence and nature of the satisfiable-to-unsatisfiable transition, observed upon decreasing the mean vertex degree of different sparse random graphs. In the replica-symmetry assumption, the authors of the cited paper study average macroscopic properties of a given random graph ensemble, making use of a numerical method of the population dynamics type. In the current work, we mainly focus on the algorithmic properties of the message-passing procedure and related decimation strategies. In particular, we discuss both analytical and numerical strategies for limiting the increase of computational cost arising from the pairwise messages. Also, in the last part of this paper, we develop the distributional version of the message-passing scheme, which in the literature is usually denoted as the cavity method [16] and used to study random (glass-like) systems [17]. We limit this analysis to the replica-symmetry assumption and to the simple colour-symmetric (paramagnetic) solution. Within these simplifying hypotheses, we compute the quenched entropy for given random graph ensembles and estimate the corresponding satisfiability threshold, partially recovering a result of [3].

2. Statement of the problem and belief propagation

We consider an undirected simple graph, whose vertices are denoted by $i = 1, \ldots, N$. Our goal is to assign to each vertex i a colour x_i from a given colour set $C \equiv \{1, 2, \ldots, q\}$, in a way to minimize the cost (energy) function:

$$E(x_1, \ldots, x_N) = \sum_{i=1}^{N} \eta(x_i, x_{\partial i}), \tag{1}$$

where ∂i denotes the neighbourhood of i (i.e. the set of vertices directly connected to i by an edge) and $x_{\partial i} \equiv \{x_j\}_{j \in \partial i}$ is the array of colour variables in ∂i. The elementary energy term $\eta(x_i, x_{\partial i})$ counts the number of missing colours in the neighbourhood of i, including i itself. A suitable expression for the η function is therefore

$$\eta(x_1, \ldots, x_n) = \sum_{x \in C} \prod_{i=1}^{n} [1 - \delta(x_i, x)], \tag{2}$$

where $\delta(x, y)$ is a Kronecker delta and n is the number of entries of the η function (not *a priori* fixed). With the above definitions, the cost function value is $E(x_1, \ldots, x_N) = 0$ if and only if the colour assignments $x_1, \ldots, x_N$ satisfy all constraints.

In the current work, we deal with this problem by studying an equivalent 'thermodynamic' system, whose potential energy is defined by the cost function $E(x_1, \ldots, x_N)$. For energy minimization, we consider the zero-temperature limit. The BP approach allows us to determine approximate marginals of the equilibrium (Boltzmann) probability distribution for the colour variables. As mentioned in section 1, our approximation becomes exact when the graph is a tree. From the treatment described in appendix A, it turns out that we can write two different marginals, namely the joint distribution of two colour variables on a graph edge $p_{i,j}(x_i, x_j)$ and the joint distribution of a given colour variable together with its neighbours $p_{i,\partial i}(x_i, x_{\partial i})$ ('cluster' distribution), as a function of pairwise messages $m_{j\to i}(x_j, x_i)$. Each given term $m_{j\to i}(x_j, x_i)$ may be viewed as a message sent from the cluster $\{j, \partial j\}$ to the edge $\{i, j\}$, representing the influence of the constraint associated with the vertex j onto the colour variables of the edge $\{i, j\}$ (some details about this interpretation are elucidated in appendix B). In formulae, we have

$$p_{i,j}(x_i, x_j) = \mathrm{e}^{f_{ij}}\, m_{i\to j}(x_i, x_j)\, m_{j\to i}(x_j, x_i), \tag{3}$$

$$p_{i,\partial i}(x_i, x_{\partial i}) = \mathrm{e}^{f_i - \beta\eta(x_i, x_{\partial i})} \prod_{j \in \partial i} m_{j\to i}(x_j, x_i), \tag{4}$$

where β is the inverse temperature, and f_{ij} and f_i, usually called free energy shifts (see appendix A), can be determined by normalization as

$$\mathrm{e}^{-f_{ij}} = \sum_{x_i,x_j} m_{i\to j}(x_i,x_j)\, m_{j\to i}(x_j,x_i), \tag{5}$$

$$\mathrm{e}^{-f_i} = \sum_{x_i,x_{\partial i}} \mathrm{e}^{-\beta\eta(x_i,x_{\partial i})} \prod_{j\in\partial i} m_{j\to i}(x_j,x_i). \tag{6}$$

The messages have to satisfy a set of self-consistency equations, which basically account for compatibility between 'overlapping' distributions. For instance, the $\{i,j\}$ edge distribution must be a marginal of the cluster distributions associated with both vertices i and j. Considering the former case, we can write

$$p_{i,j}(x_i,x_j) = \sum_{x_{\partial i\setminus j}} p_{i,\partial i}(x_i,x_{\partial i}), \tag{7}$$

where the sum runs over the values of the array of colour variables $x_{\partial i\setminus j} \equiv \{x_k\}_{k\in\partial i\setminus j}$, i.e. the colour variables in the neighbourhood of i except x_j. In fact, we can obtain the self-consistency equation by replacing (3) and (4) into the compatibility equation (7), yielding

$$m_{i\to j}(x_i,x_j) \propto \sum_{x_{\partial i\setminus j}} \mathrm{e}^{-\beta\eta(x_i,x_{\partial i})} \prod_{k\in\partial i\setminus j} m_{k\to i}(x_k,x_i), \tag{8}$$

where a normalization factor has been replaced by the proportionality symbol. In order to satisfy all the necessary compatibilities, one equation of the above form must hold for each directed edge $i\to j$. The BP algorithm solves the set of self-consistency equations iteratively, starting from suitable (usually random or uniform) initial conditions for the messages, until the distance between messages at subsequent updates goes below a given threshold. From a heuristic point of view, each message update according to (8) is usually interpreted as a propagation process, so that in the following we shall also denote (8) as the propagation equation. For completeness, in appendix B we also report the propagation equations of the naive BP algorithm, which are numerically simpler.

We note that, by employing the explicit expression (2) of the elementary energy term (cluster energy), we can significantly reduce the computational cost of the propagation equation (8) as well. Indeed, it turns out that the latter can be rewritten as

$$m_{i\to j}(x_i,x_j) \propto \sum_{B\subseteq C\setminus x_i\setminus x_j} (-1+\mathrm{e}^{-\beta})^{|B|} \prod_{k\in\partial i\setminus j} \sum_{x_k\in C\setminus B} m_{k\to i}(x_k,x_i), \tag{9}$$

where the outer sum runs over all the possible subsets B of the colour set C without the colours x_i, x_j. The derivation can be found in appendix C. Now, we compare the computational cost of the generic equations with respect to the simplified form. Assuming that d is the degree of vertex i, the generic equation (8) requires $(d-1)q^{d-1}$ multiplications, which can be reduced to $2q^{d-1} + \sum_{n=2}^{d-2} q^{d-n}$ by suitable (straightforward) programming tricks. Taking into account that a trivial necessary condition for an elementary constraint to be satisfiable is $d \geq q-1$, the leading term of the computational cost turns out to be at least q^{q-2}. The simplified equation (9), however, requires $(d-1)2^{q-2}$ multiplications, which is clearly much more convenient for any $q > 2$.

Finally (for completeness and future use), we also report the simplified expression of the cluster free energy shift (6):

$$\mathrm{e}^{-f_i} = \sum_{x_i \in C} \sum_{B \subseteq C \setminus x_i} (-1 + \mathrm{e}^{-\beta})^{|B|} \prod_{j \in \partial i} \sum_{x_j \in C \setminus B} m_{j \to i}(x_j, x_i), \tag{10}$$

which can be obtained by an analogous derivation.

3. Optimization strategy and numerical results

In this section, we define the optimization strategy and test its performance on single instances of random graphs drawn from a suitable ensemble. Our strategy involves a decimation procedure, which is analogous to that of [2], but is carried out on the basis of nearest-neighbour pair distributions $p_{i,j}(x_i, x_j)$, rather than single-variable distributions. Given a graph and a number q of available colours, we first fix the colour of a randomly chosen vertex in order to break the colour permutation symmetry, and proceed as follows. We perform the first BP run (starting from uniform messages) and determine the pair distributions according to (3). For each edge $\{i, j\}$, we fix the colour variables x_i, x_j at the values $\bar{x}_i, \bar{x}_j$ having the largest joint probability, provided the latter is larger than a certain threshold. If no probability satisfies such a condition, we only fix the pair of variables with the largest joint probability over the whole graph. Then, we rerun BP (starting from the previously computed messages) and iterate the above procedure until all variables are fixed or all constraints are satisfied (in the latter case, non-fixed variables can be assigned a random colour). We always set the threshold probability at 0.9, as done in [2]. We observe that, in most cases, one of the two variables chosen to be fixed has already been fixed at a previous stage of the decimation procedure, so that, in most cases, we actually fix just one variable for each given pair. Therefore, even though we are working with pair, rather than single-variable distributions, we observe that choosing the same threshold probability results in a similar decimation rate.

We now spend a few words on the precise meaning of 'fixing a variable', as introduced above, from the point of view of the message-passing procedure. In the thermodynamic language, colouring a vertex is tantamount to imposing an infinite energy penalty to all other possible colours. Thus, if we want to fix a single variable x_i to a given colour $\bar{x}_i$, we may add to the corresponding cluster energy $\eta(x_i, x_{\partial i})$ a term $\gamma[1 - \delta(x_i, \bar{x}_i)]$, and then take the limit $\gamma \to \infty$. By the propagation equation (8), it is easy to see that such operations imply that all the messages $m_{i \to j}(x_i, x_j)$, sent from the vertex i (more precisely, from the cluster associated with the vertex i) must be multiplied by a prefactor $\delta(x_i, \bar{x}_i)$, which basically preserves only messages of the type $m_{i \to j}(\bar{x}_i, x_j)$. As a consequence, when we fix the colours of two nearby vertices, it turns out that the latter no longer need to exchange messages or, in other words, the messages remain fixed at

$$m_{i \to j}(x_i, x_j) = m_{j \to i}(x_j, x_i) = \delta(x_i, \bar{x}_i)\, \delta(x_j, \bar{x}_j). \tag{11}$$

Although such messages have no effect on the vertices i and j themselves, due to the form of the propagation equation, they may still influence their neighbourhoods $\partial i \setminus j$ and $\partial j \setminus i$.

Before presenting the results, we note that in [2] the authors observe that the naive BP hardly ever converges. This problem is circumvented by computing probability distributions as 'time averages' over a number of iterations, which turns out to provide sufficient information for guiding the decimation procedure. In our scheme, the BP algorithm turns out to converge more frequently, except in the vicinity of the satisfiability threshold (especially after several vertices have been coloured). Convergence may be improved by computing the message updates as convex linear combinations between the old estimates (with coefficient α) and the updates obtained from the propagation equation (with coefficient $1-\alpha$). The adjustable parameter α plays the role of a damping in the propagation dynamics and we refer to it as the damping parameter. Nevertheless, we generally find that reaching convergence is not really necessary. Indeed, a very small number ν of sequential updates[7] of all messages are sufficient to provide the relevant information about pair probabilities and that a larger number of iterations does not significantly improve the overall algorithm performance. This fact allows us to drastically reduce the computational cost of the full procedure, although it does not affect the complexity of a single iteration.

We are now in a position to perform a quantitative comparison with the naive BP approach [2]. As in the cited work, we consider a number of available colours $q = 4$ and random graphs with $N = 1000$ vertices. Graphs are generated in such a way to have vertices with two different degrees $d = \lfloor c \rfloor$ and $d = \lceil c \rceil$, where c is the mean degree. The degree distribution, i.e. the probability of a vertex having degree d, is therefore

$$\rho_d = \begin{cases} \lceil c \rceil - c & \text{if } d = \lfloor c \rfloor \\ c - \lfloor c \rfloor & \text{if } d = \lceil c \rceil \\ 0 & \text{otherwise,} \end{cases} \tag{12}$$

which we denote as a linear distribution. We always assume $c \geq q-1$, in order to avoid the appearance of vertices with degree less than $q-1$, for which the local constraints are necessarily unsatisfiable. We do not report results about graphs with cut-Poissonian degree distribution [2], which exhibit analogous behaviour.

In figure 1 we report both perfect colouring and unsatisfaction measures, over 1000 random graph samples, as a function of the mean degree. The perfect colouring measure is simply defined as the fraction of samples for which the algorithm has been able to find a colour assignment satisfying all constraints. The unsatisfaction measure counts the fraction of missing colours per vertex, i.e. the energy per vertex divided by the total number of colours, $E(x_1, \ldots, x_N)/Nq$ ($x_1, \ldots, x_N$ being the colour assignments found by the algorithm), averaged over all samples. We can see that the BP approach improves the naive one in both respects. The perfect colouring measure turns out to be consistently increased in the vicinity of the critical mean degree values, below which it rapidly vanishes. In this region, naive BP itself was already found to work better than the Walksat-like algorithm analysed in [2].

In analogy with the ordinary colouring problem [18] (though with a reversed role for the mean degree c), we expect that, for even lower c values, our problem becomes unsatisfiable with high probability (i.e. with probability tending to 1 in the 'thermodynamic' $N \to \infty$ limit). We also expect the presence of an intermediate hard-satisfiable phase in which the problem is satisfiable with high probability but BP fails, because of a clustered structure of the solution space (replica-symmetry breaking) [16]–[20]. Accordingly, the perfect colouring probability falling down to zero is likely to indicate

[7] With reference to the propagation equation (8), by sequential update we mean that, in generating a given 'output' (left-hand side) message, one makes use of updated 'input' (right-hand side) messages, if already available.

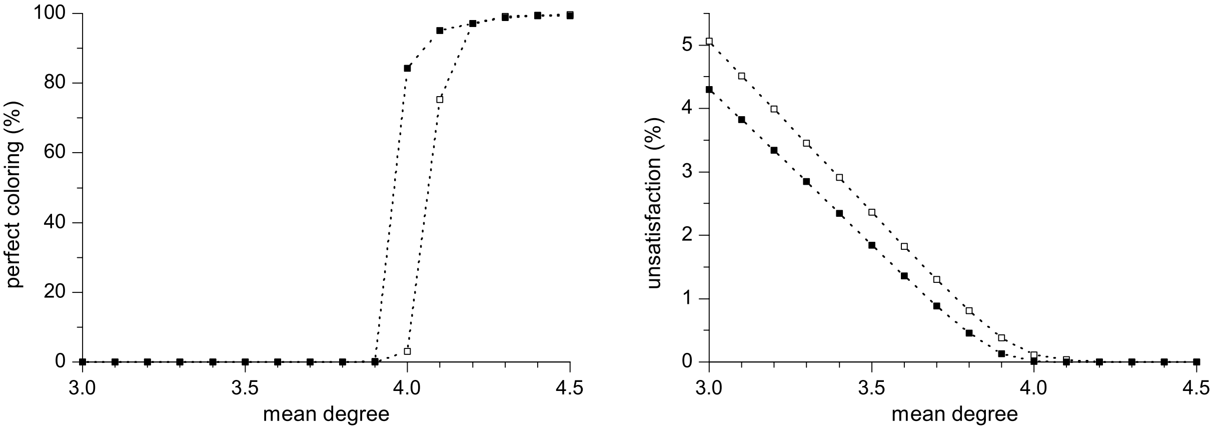

Figure 1. Perfect colouring (left) and unsatisfaction (right) measures over 1000 graphs for naive BP [2] (open squares) and BP with $\nu = 3$ and $\alpha = 0.1$ (solid squares), as a function of the mean degree. In both cases the inverse temperature used for the computation is $\beta = 10$.

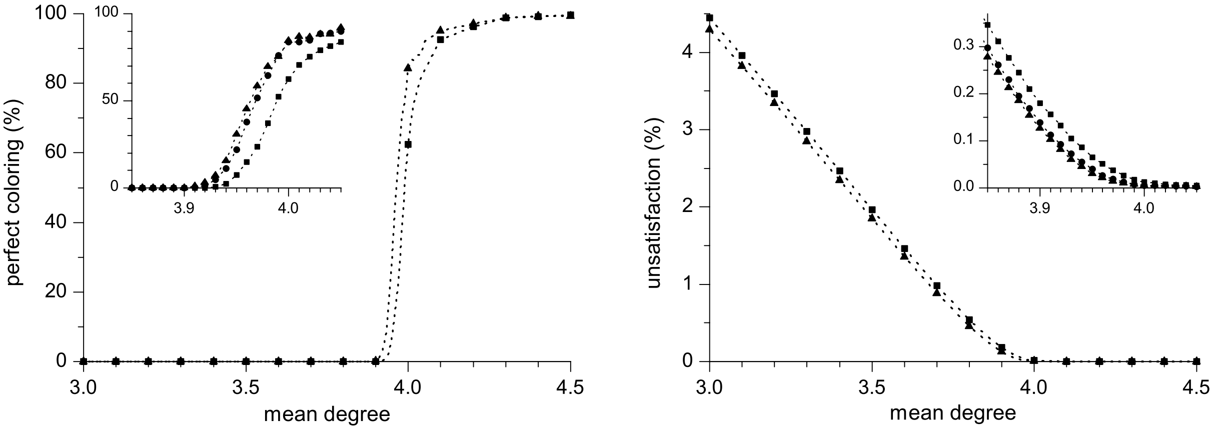

Figure 2. Perfect colouring (left) and unsatisfaction (right) measures over 1000 graphs for BP with $\alpha = 0.1$ and $\beta = 10$ as a function of the mean degree. Squares, circles and triangles denote $\nu = 1, 2$ and 3, respectively. In the main figures, interpolation between data points in the transition region has been performed by taking into account the extra data points reported in the insets.

the onset of such hard-satisfiable phase rather than the truly unsatisfiable phase. We shall return to this point later. For the moment, we observe that the BP approach definitely works better than the naive one, even for very low c values, in the (expected) unsatisfiable phase. In this region we observe both a reduction of the unsatisfaction measure itself and of its growth rate with decreasing c.

Concerning the percentage of perfect colouring, we have noticed that the performance of the algorithm is significantly affected by the number ν of iterations per decimation step, only in a narrow region close to the critical c value. This suggests that in this region the problem is actually more difficult to solve. Some results about the influence of the ν parameter are reported in figure 2. Upon increasing ν, some improvement can also be

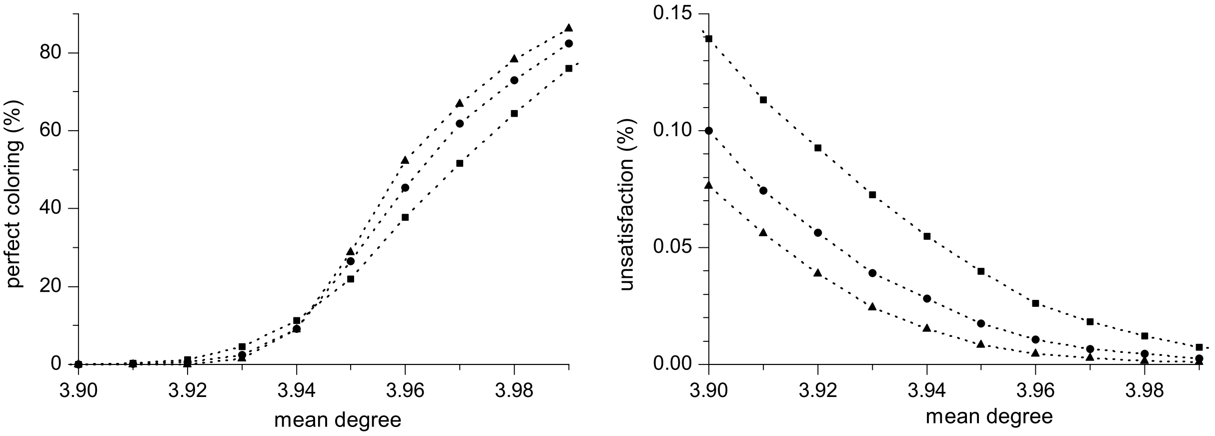

Figure 3. Perfect colouring (left) and unsatisfaction (right) measures over 1000 graphs for BP with $\nu = 2$, $\alpha = 0.1$ and $\beta = 10$, as a function of the mean degree. Squares, circles and triangles denote number of vertices $N = 1000, 2000$ and 4000, respectively.

observed in the unsatisfaction measure. However, as previously mentioned, increasing ν values beyond 2 or 3 does not yield any further significant improvement. We also note that a quantitatively comparable improvement of the unsatisfaction measure is obtained by choosing a small but nonzero value of the damping parameter α. All the results reported in the current paper have been obtained with $\alpha = 0.1$, but it turns out that in a rather large range ($0.03 \lesssim \alpha \lesssim 0.3$) the average algorithm performance is practically independent of the precise value of the damping parameter. Finally, we note that (for $\nu \geq 2$) the perfect colouring measure exhibits a slight kink at $c = 4.0$. This can be ascribed to an abrupt change in the structure of the graph ensemble. In fact, according to the linear degree distribution (12), for $c = 4$ all vertices have exactly degree 4, whereas for $c > 4$ or $c < 4$ a few vertices appear with degree 5 or 3, respectively.

We have also analysed the algorithm behaviour as a function of the number of vertices N. The results are reported in figure 3. We can see that the transition in the perfect colouring probability becomes more and more abrupt upon increasing N, and a crossover point appears at a mean degree value $c \approx 3.943$. Even though the precise crossover value may depend on the algorithm parameters, such behaviour suggests that the transition may be sharp (first-order-like) in the $N \to \infty$ limit. The latter conjecture is consistent with the fact that random graphs of increasing size become more and more tree-like, such that the BP approach is able to provide better and better approximations. In principle, the crossover point might be the signature of the satisfiable-to-unsatisfiable transition but, as previously mentioned, we are rather led to identify it with the onset of the hard-satisfiable phase. Indeed, the estimate of the satisfiability threshold, carried out in the next section, provides further evidence in favour of the latter hypothesis.

4. Entropy and satisfiability threshold

In this section, we study average macroscopic properties of the BP solution over random graph ensembles, with particular attention to the average entropy. The latter is usually denoted as quenched entropy in statistical mechanics language. Taking the limit $\beta \to \infty$, this quantity provides an average measure of (the logarithm of) the number of zero energy configurations, i.e. perfect colourings, for a given ensemble, which also allows us to estimate the satisfiability threshold. In this context, the main source of approximation will be the replica-symmetry assumption, since the approximation due to BP itself is expected to be negligible in the infinite size limit. Furthermore, we limit the analysis to BP solutions that do not break the colour permutation symmetry ('paramagnetic' solutions), because we have numerical evidence that, when BP converges, no spontaneous symmetry breaking of the solution is ever observed. Average properties of non-paramagnetic (glass-like) solutions have been investigated in [3], but they only appear at very low c values, where the replica-symmetry assumption is expected to break down anyway.

According to the paramagnetic ansatz, the messages are always such that $m_{i\to j}(x,x)$ does not depend on x, and $m_{i\to j}(x,y)$ does not depend on x,y, if $x \neq y$. This means that the only important quantity is $u_{i\to j} \equiv m_{i\to j}(x,x)/m_{i\to j}(x,y)$, i.e. the ratio between the 'equal colours' message and the 'different colours' message. Taking into account that the message normalization is irrelevant to all observable quantities, we can write the full message as

$$m_{i\to j}(x,y) = 1 - (1-u_{i\to j})\,\delta(x,y) = \begin{cases} u_{i\to j} & \text{if } x = y \\ 1 & \text{otherwise.} \end{cases} \tag{13}$$

We note that in principle one could also think about the inverse ratio $m_{i\to j}(x,y)/m_{i\to j}(x,x)$ as the relevant message, but this choice turns out to be unfeasible, due to the nature of the constraints, favouring the presence of different neighbouring colours. Indeed, at zero temperature, it is easy to foresee the emergence of 'hard' messages such that $m_{i\to j}(x,x) = 0$, stemming from vertices with degree $q-1$ (whose all-neighbours are forced to have a different colour), whereas we always expect $m_{i\to j}(x,y) \neq 0$ for $x \neq y$ in a paramagnetic state.

Replacing (13) into the inner sum appearing in the simplified propagation equation (9), we can write

$$\sum_{x_k\in C\setminus B} m_{k\to i}(x_k,x_i) = q - |B| - 1 + u_{k\to i}, \tag{14}$$

where the term $-1+u_{k\to i}$ appears because $x_i \notin B$. Since the sum above only depends on B via its cardinality $|B|$, in (9) we can replace the sum over B by a sum over cardinalities, inserting suitable binomial coefficients. Thus we finally obtain a reduced propagation equation for the message ratios:

$$u_{i\to j} = \frac{\sum_{n=0}^{q-1}\binom{q-1}{n}(-1)^n \prod_{k\in\partial i\setminus j}(q-n-1+u_{k\to i})}{\sum_{n=0}^{q-2}\binom{q-2}{n}(-1)^n \prod_{k\in\partial i\setminus j}(q-n-1+u_{k\to i})}, \tag{15}$$

in which we have also taken the zero temperature ($\beta \to \infty$) limit. The cluster free energy shift can be similarly derived by replacing (13) in (10). In the zero temperature limit, we obtain

$$\mathrm{e}^{-f_i} = q\sum_{n=0}^{q-1}\binom{q-1}{n}(-1)^n \prod_{j\in\partial i}(q-n-1+u_{j\to i}). \tag{16}$$

The edge free energy shifts can be directly obtained by inserting (13) into (5):

$$\mathrm{e}^{-f_{ij}} = q\,(q-1+u_{i\to j}\,u_{j\to i}). \tag{17}$$

We can characterize a random graph ensemble by a probability distribution of messages $P(u)$. Such a distribution has to obey a functional equation (usually known as the cavity equation [16]) of the following form:

$$P(u) = \sum_d \tilde{\rho}_d \int \mathrm{d}u_1\, P(u_1)\cdots \int \mathrm{d}u_{d-1}\, P(u_{d-1})\delta(u-\hat{u}(u_1,\ldots,u_{d-1})), \tag{18}$$

where $\hat{u}(u_1,\ldots,u_{d-1})$ is the 'propagation function' defined by (15), and where $\tilde{\rho}_d$ is the probability of finding a vertex of degree d by choosing a random direction in a randomly selected edge. It is easy to see that $\tilde{\rho}_d$ is related to the degree distribution ρ_d as

$$\tilde{\rho}_d = \frac{d\rho_d}{c}. \tag{19}$$

In the context of the cavity method, the replica-symmetry assumption consists in the fact that we consider a single distribution of messages. In a replica-symmetry breaking scenario, each propagated quantity $u_{i\to j}$ (message) would be replaced by a probability distribution defined over different ergodic components (states) [16].

We solve the functional equation (18) numerically by a population dynamics approach [16]. In a nutshell, we represent the distribution $P(u)$ by an evolving population of messages. An elementary evolution step consists in generating a new message according to the propagation equation (15), making use of $d-1$ messages randomly taken from the population, where d is randomly generated according to the $\tilde{\rho}_d$ distribution. The newly generated message replaces a randomly selected message of the population. Due to the presence of hard messages $u=0$ generated by degree $q-1$ vertices, we observe that the message distribution $P(u)$ contains a Dirac delta peak centred in zero with weight $\tilde{\rho}_{q-1}$.

From the message distribution, we can evaluate the average cluster and edge free energy shifts as

$$\overline{f_{\mathrm{c}}} = \sum_d \rho_d \int \mathrm{d}u_1 P(u_1)\cdots \int \mathrm{d}u_d\, P(u_d) f_{\mathrm{c}}(u_1,\ldots,u_d), \tag{20}$$

$$\overline{f_{\mathrm{e}}} = \int \mathrm{d}u_1\, P(u_1) \int \mathrm{d}u_2\, P(u_2) f_{\mathrm{e}}(u_1,u_2), \tag{21}$$

where the functions $f_{\mathrm{c}}(u_1,\ldots,u_d)$ and $f_{\mathrm{e}}(u_1,u_2)$ are defined by (16) and (17). Thus we obtain the average free energy per vertex as

$$\bar{f} = \overline{f_{\mathrm{c}}} - \frac{c}{2}\overline{f_{\mathrm{e}}}, \tag{22}$$

where $c/2$ is the average number of edges per vertex. The above formula directly descends from (A.12). Finally, since we have incorporated a β factor in our free energy definition, and since the limit $\beta\to\infty$ fixes the energy at zero, the entropy per vertex is simply $s=-\bar{f}$.

For actual calculations, we have considered random graph ensembles with the linear degree distribution (as defined in section 3), and with the cut-Poissonian distribution

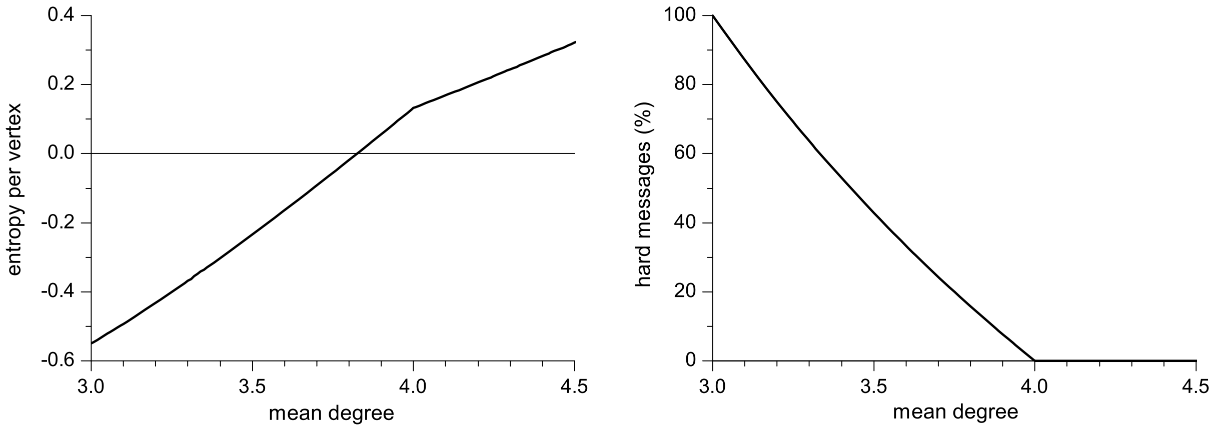

Figure 4. Entropy per vertex (left) and fraction of hard messages (right) for random graphs with linear degree distribution ($q = 4$), as a function of the mean degree c.

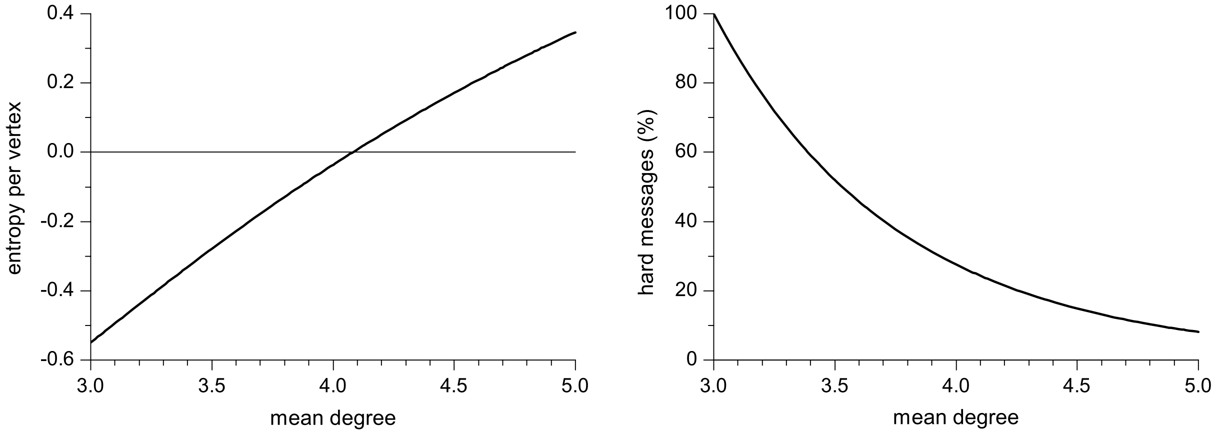

Figure 5. The same as figure 4 for random graphs with cut-Poissonian degree distribution.

(also considered in [2]), defined as

$$\rho_d = \begin{cases} \mathrm{e}^{-(c-q+1)} \dfrac{(c-q+1)^{d-q+1}}{(d-q+1)!} & \text{if } d \geq q-1 \\ 0 & \text{otherwise,} \end{cases} \tag{23}$$

where c is still the mean degree. This distribution also excludes vertices with degree smaller than $q-1$ and, hence, trivially unsatisfiable constraints.

In figures 4 and 5 we report the results for the two ensembles, respectively. As expected, the entropy turns out to be a monotonically increasing function of the mean degree, as the problem becomes easier to satisfy. It is interesting to note that the fraction of hard messages $\tilde{\rho}_{q-1}$ for the linear degree distribution turns out to be nonzero only for $c < q$, which explains the kink observed in the entropy function. Negative entropy identifies the unsatisfiable region (perfect colourings are exponentially rare), whereas the zero entropy point identifies the satisfiability threshold c_{th}. For the two ensembles, we respectively find $c_{\mathrm{th}} \approx 3.825$ and $c_{\mathrm{th}} \approx 4.082$. As previously mentioned, we expect that these values are, in fact, approximate ones, because we have neglected the possibility of replica-symmetry breaking. Nevertheless, these values are in reasonable agreement with the numerical estimates put forward in [2], namely $c_{\mathrm{th}} \approx 3.8$ and $c_{\mathrm{th}} \approx 4.1$. As far as the linear ensemble is concerned, we expect that our result is also analytically equivalent to the (replica-symmetric) one by Wong and Saad [3], and in fact we obtain a very good numerical agreement for the threshold value.

5. Summary and conclusions

In this paper, we have considered a variation of the well-known graph colouring problem, which may be viewed as the prototype of a combinatorial optimization problem emerging in the context of distributed data storage. We have worked out the BP equations for this problem, which provide the exact solution on a tree. Due to the many-body nature of the problem, such equations turn out to be different from the naive BP message-passing scheme, as the latter involves messages sent to single variables, whereas the former involve messages sent to pairs of nearest-neighbour variables. Our simulations, performed on random graphs drawn from a suitable ensemble, suggest that the new algorithm, associated with a decimation procedure, turns out to be much more effective than the naive BP-based algorithm. In particular, the probability of finding a perfect colouring is significantly enhanced, especially in the vicinity of the satisfiable-to-unsatisfiable transition. Furthermore, both the unsatisfaction measure and its growth rate upon decreasing the average graph connectivity are significantly reduced. This improved performance is, however, obtained at the cost of increased computational complexity. Therefore, we have suggested two possible ways of reducing this complexity. On the one hand, we have shown some analytical manipulations (exploiting the particular form of the constraints) can simplify a single iteration. On the other hand, numerical experiments have shown that very few iterations (even a single one) provide sufficient information to drive the decimation procedure. We note that, in this way, our decimation procedure turns out to be somehow 'distributed' over different BP iterations. This fact partially reminds us of the so-called 'reinforced BP' approach, which has been successfully exploited to solve different combinatorial optimization problems [21]. Although beyond the scope of the current paper, it might be interesting to analyse the performance of the latter method for the palette-colouring problem. Indeed, the reinforcement strategy would replace the decimation procedure, allowing for a fully decentralized implementation of the algorithm, which might make it actually appealing from a practical/technical point of view.

From a more theoretical perspective, we have applied the cavity method to investigate the satisfiable-to-unsatisfiable transition, which appears upon decreasing the average graph connectivity. Limiting this analysis to the replica-symmetry assumption, we have observed that the threshold connectivity seems to be significantly displaced with respect to the value observed in the numerical experiments. As previously mentioned, this fact suggests that the breakdown of the algorithm may occur because of the onset of a hard-satisfiable phase. It would also be interesting to investigate this possibility, making use of the cavity method at the level of one-step replica-symmetry breaking [16], along the lines of several works dealing with the ordinary colouring problem [18]–[20].

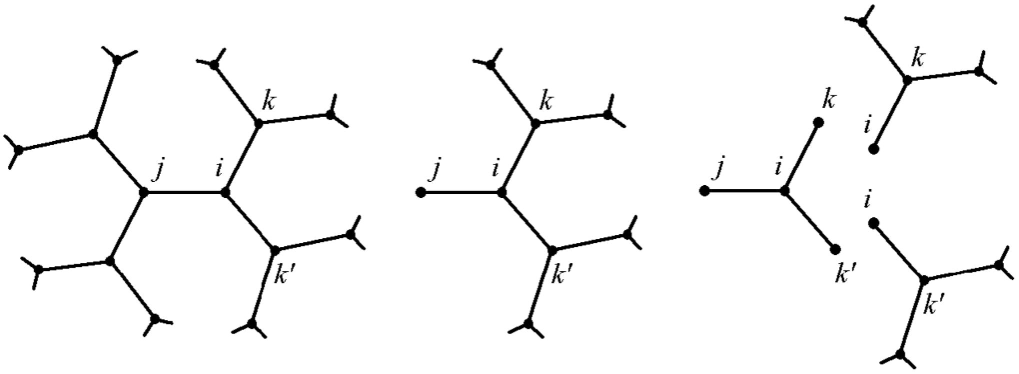

Figure A.1. Tree graph (left), disconnected branch $i \to j$ (centre) and decomposition of the latter into subbranches $k \to i$, for $k \in \partial i \setminus j$, plus the elementary cluster associated with i (right).

Appendix A. Belief propagation equations

The BP equations can in general be derived from a very simple recipe. One first ‘fakes’ that the graph is a tree and then formally applies the equations obtained for such a case to a generic graph. This derivation also provides a heuristic argument explaining why the method generally works better for graphs with a tree-like structure.

According to the Boltzmann law, the joint probability distribution of all the colour variables can be written as

$$p(x_1, \ldots, x_N) = \mathrm{e}^{F-\beta E(x_1,\ldots,x_N)}, \tag{A.1}$$

where $E(x_1, \ldots, x_N)$ is the energy function (1), β is the inverse temperature, and F is the free energy (times β), which can be determined by normalization. Following our ‘fake assumption’, we can consider, for each edge $i \to j$ (defined with a direction), the branch growing from the root vertex j towards i, disconnected from the remainder of the system (see figure A.1). We can thus define a partial energy function $E_{i\to j}(x_{i\to j})$, obtained by summing the elementary interaction energies only for vertices in the branch, except the root vertex. Since our elementary interaction energies couple together clusters of variables including each vertex and all its neighbours, each partial energy function depends on the array of all colour variables in the branch including the root vertex. We denote this array by $x_{i\to j}$. Now, each disconnected branch can be ideally studied as an independent subsystem, whose Boltzmann probability distribution turns out to be

$$p_{i\to j}(x_{i\to j}) = \mathrm{e}^{F_{i\to j}-\beta E_{i\to j}(x_{i\to j})}, \tag{A.2}$$

where $F_{i\to j}$ denotes the corresponding free energy. Note that it is possible to decompose the partial energy of the given branch $i \to j$ into a sum of the partial energies of its subbranches $k \to i$, for all $k \in \partial i \setminus j$, plus the elementary interaction energy associated with i (see figure A.1):

$$E_{i\to j}(x_{i\to j}) = \eta(x_i, x_{\partial i}) + \sum_{k\in\partial i\setminus j} E_{k\to i}(x_{k\to i}). \tag{A.3}$$

doi:10.1088/1742-5468/2011/05/P05010

We also define a free energy shift $f_{i\to j}$ as the difference between the free energy of the $i \to j$ disconnected branch and the sum of free energies of its (disconnected) subbranches, i.e.

$$F_{i\to j} = f_{i\to j} + \sum_{k\in\partial i\setminus j} F_{k\to i}. \tag{A.4}$$

From (A.2)–(A.4), we can write

$$p_{i\to j}(x_{i\to j}) = \mathrm{e}^{f_{i\to j}-\beta\eta(x_i,x_{\partial i})} \prod_{k\in\partial i\setminus j} p_{k\to i}(x_{k\to i}), \tag{A.5}$$

which provides a relationship between the Boltzmann distribution of the $i \to j$ disconnected branch and those of its (disconnected) subbranches. Defining the messages $m_{i\to j}(x_i,x_j)$ as marginals of a corresponding branch distribution $p_{i\to j}(x_{i\to j})$ over the variables x_j and x_i (respectively, the root vertex and its first neighbour in the branch) we finally obtain the self-consistency equation (8).

We still have to show how messages can determine cluster and edge marginals of the full Boltzmann distribution (A.1). As in our previous manipulations, we observe that, for each given vertex i, it is possible to write the total energy function (1) as a sum of partial energies of the disconnected branches $j \to i$, for all $j \in \partial i$, plus the elementary interaction energy associated with i:

$$E(x_1,\ldots,x_N) = \eta(x_i,x_{\partial i}) + \sum_{j\in\partial i} E_{j\to i}(x_{j\to i}). \tag{A.6}$$

Defining also the free energy shift f_i as the difference between the total free energy F and the sum of the disconnected branch free energies, for all the possible branches growing from vertex i, i.e.

$$F = f_i + \sum_{j\in\partial i} F_{j\to i}, \tag{A.7}$$

from (A.1), (A.2), (A.6) and (A.7), we easily obtain

$$p(x_1,\ldots,x_N) = \mathrm{e}^{f_i-\beta\eta(x_i,x_{\partial i})} \prod_{j\in\partial i} p_{j\to i}(x_{j\to i}). \tag{A.8}$$

Now, the cluster distribution $p_{i,\partial i}(x_i,x_{\partial i})$ for each vertex i can be derived as a suitable marginal of $p(x_1,\ldots,x_N)$. By this marginalization, we obtain (4). As far as edge marginals are concerned, we have to consider a different decomposition of the total energy function. Namely, for each edge $\{i,j\}$, the former can be written as a sum of two contributions from respectively the branch starting from j towards i and the one starting from i towards j:

$$E(x_1,\ldots,x_N) = E_{i\to j}(x_{i\to j}) + E_{j\to i}(x_{j\to i}). \tag{A.9}$$

We define the free energy shift f_{ij} as the difference between the total free energy F and the sum of the free energies of the disconnected branches mentioned above, i.e.

$$F = f_{ij} + F_{i\to j} + F_{j\to i}. \tag{A.10}$$

From (A.1), (A.2), (A.9) and (A.10), we obtain

$$p(x_1,\ldots,x_N) = \mathrm{e}^{f_{ij}} p_{i\to j}(x_{i\to j}) p_{j\to i}(x_{j\to i}). \tag{A.11}$$

Evaluating the edge distribution $p_{i,j}(x_i,x_j)$ as a marginal of $p(x_1,\ldots,x_N)$, we obtain (3).

Finally, we determine the total free energy as a function of the free energy shifts. First we sum both sides of (A.7) over all vertices i and both sides of (A.10) over all edges $\{i, j\}$. Then we subtract the latter equation from the former. It is easy to see that, on a tree, the number of vertices equals the number of edges plus one, such that the left-hand side of the resulting equation turns out to be exactly F. Furthermore, in the right-hand side all the branch free energies cancel out, and we obtain

$$F = \sum_{i=1}^{N} f_i - \sum_{\{i,j\}} f_{ij}, \tag{A.12}$$

where $\sum_{\{i,j\}}$ denotes the sum over all edges.

Appendix B. Factor graph formalism

In this appendix, we first introduce a more general form of BP equations, defined on a factor graph [22]. Then, we show that from this form one can derive both the naive BP equations of [2] and the BP equations of the current paper by two different factor graphs associated with the same problem.

A factor graph is a bipartite graph, whose left- and right-side vertices are usually referred to as variable nodes and function nodes. The notion of factor graph is meant to describe the structure of the energy function, whose independent variables (i.e. the configuration variables of the corresponding thermodynamic system) are associated with the variable nodes. A function node connected to a number of variable nodes represents an elementary interaction among the corresponding variables. Let V denote the set of all the variable nodes, such that each node $v \in V$ is associated with a configuration variable x_v. Let also $A \subseteq V$ denote any subset (cluster) of variable nodes and let $x_A \equiv \{x_v\}_{v\in A}$ denote the array of the associated configuration variables. We can thus write the energy function as

$$E(x_V) = \sum_{A\in\mathcal{F}} \epsilon_A(x_A), \tag{B.1}$$

where $\epsilon_A(x_A)$ denotes the elementary interaction energy among the variables in the cluster A (cluster energy), whereas the sum runs over the set $\mathcal{F}$ of all the interacting clusters. In what follows, the same label A denotes both a function node and the cluster of variable nodes connected to it. An example of factor graphs describing the energy function of a palette-colouring problem is sketched in figure B.1.

When the factor graph is a tree, an argument similar to that in appendix A allows one to write marginals of the Boltzmann distribution as follows:

- For each variable node $v \in V$ we have the marginal:

$$p_v(x_v) = \mathrm{e}^{f_v} \prod_{\substack{A\in\mathcal{F} \\ A\ni v}} m_{A\to v}(x_v), \tag{B.2}$$

where the product runs over all the clusters A to which v belongs (i.e. all the function nodes connected to v), $m_{A\to v}(x_v)$ is a function-to-variable message and f_v is a free energy shift (ensuring normalization).

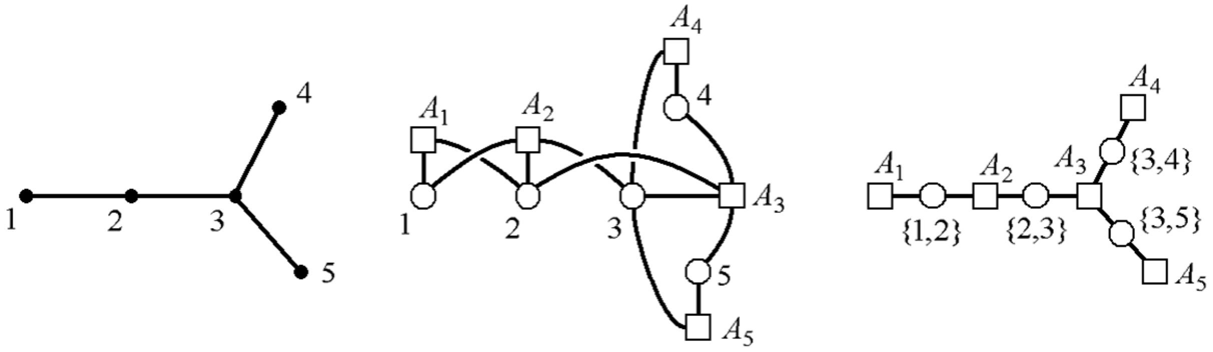

Figure B.1. A simple undirected graph (left) and the related factor graphs giving rise to naive BP (centre) and BP (right). Open circles and squares denote variable and function nodes, respectively. The labels are explained in the text.

- For each cluster $A \in \mathcal{F}$, we have the marginal:

$$p_A(x_A) = \mathrm{e}^{f_A - \beta\epsilon_A(x_A)} \prod_{v\in A} w_{v\to A}(x_v), \tag{B.3}$$

where f_A is a free energy shift, and where $w_{v\to A}(x_v)$ is a variable-to-function message:

$$w_{v\to A}(x_v) = \prod_{\substack{A'\in\mathcal{F}\setminus A\\ A'\ni v}} m_{A'\to v}(x_v), \tag{B.4}$$

a product of the messages sent to v from all connected function nodes except A.

As shown in section 2, one can derive the propagation equations by imposing compatibility between overlapping distributions. In this case, for all $A \in \mathcal{F}$ and for all $v \in A$, we can write

$$p_v(x_v) = \sum_{x_{A\setminus v}} p_A(x_A), \tag{B.5}$$

where the sum runs over all possible values of the variables in the cluster A except x_v. Inserting (B.2) and (B.3) into (B.5), we obtain the propagation equation

$$m_{A\to v}(x_v) \propto \sum_{x_{A\setminus v}} \mathrm{e}^{-\beta\epsilon_A(x_A)} \prod_{v'\in A\setminus v} w_{v'\to A}(x_{v'}), \tag{B.6}$$

with the $w_{v'\to A}(x_{v'})$ defined by (B.4). Note that, as in (8), we have replaced the normalization factor with a proportionality symbol. Finally, following the argument of appendix A, we write the total free energy as a function of the free energy shifts as

$$F = \sum_{A\in\mathcal{F}} f_A - \sum_{v\in V} (d_v - 1) f_v, \tag{B.7}$$

where d_v is the degree of the variable node v in the factor graph.

Naive BP

We first consider the energy function (1), where the configuration (colour) variables x_i are associated with the vertices $i = 1, \ldots, N$ of an ordinary graph, and the elementary interaction energy involves a cluster made up of a vertex i and all its neighbours ∂i. This structure is described by a factor graph in which the variable nodes are associated with the vertices of the original graph and the function nodes with the clusters. We can use the same index for both the variable node i and the function node with i at its centre (the cluster $A_i \equiv \{i, \partial i\}$). Hence, each variable node i receives messages $m_{A_j \to i}(x_i)$ from all the function nodes A_j with $j \in \partial i$, and from A_i itself. With the short-hand $m_{j\to i}$ for $m_{A_j \to i}$, omitting the normalization factor, (B.2) becomes

$$p_i(x_i) \propto m_{i\to i}(x_i) \prod_{j\in\partial i} m_{j\to i}(x_i). \tag{B.8}$$

Similarly, a function node A_i receives variable-to-function messages from i and all $j \in \partial i$, and the cluster distribution for A_i (B.3) becomes

$$p_{i,\partial i}(x_i, x_{\partial i}) \propto \mathrm{e}^{-\beta\eta(x_i, x_{\partial i})} w_{i\to i}(x_i) \prod_{j\in\partial i} w_{j\to i}(x_j). \tag{B.9}$$

We have identified $\epsilon_{A_i}(x_{A_i})$ with $\eta(x_i, x_{\partial i})$, and $w_{j\to i}$ is short-hand for $w_{j\to A_i}$. From (B.4), one can see that the variable-to-function messages take two slightly different forms, depending on whether they travel (to the cluster A_i) either from the 'central' node i or from a 'peripheral' node $j \in \partial i$. In the simplified notation, we have respectively

$$w_{i\to i}(x_i) = \prod_{j\in\partial i} m_{j\to i}(x_i), \tag{B.10}$$

$$w_{j\to i}(x_j) = m_{j\to j}(x_j) \prod_{k\in\partial j\setminus i} m_{k\to j}(x_j). \tag{B.11}$$

The compatibility condition (B.5) can also be written in two different forms. For all $i = 1, \ldots, N$, $j \in \partial i$, we have respectively

$$p_i(x_i) = \sum_{x_{\partial i}} p_{i,\partial i}(x_i, x_{\partial i}), \tag{B.12}$$

$$p_i(x_i) = \sum_{x_j, x_{\partial j\setminus i}} p_{j,\partial j}(x_j, x_{\partial j}). \tag{B.13}$$

Using (B.8) and (B.9), this in turn gives rise to two different propagation equations:

$$m_{i\to i}(x_i) \propto \sum_{x_{\partial i}} \mathrm{e}^{-\beta\eta(x_i, x_{\partial i})} \prod_{j\in\partial i} w_{j\to i}(x_j), \tag{B.14}$$

$$m_{j\to i}(x_i) \propto \sum_{x_j, x_{\partial j\setminus i}} \mathrm{e}^{-\beta\eta(x_j, x_{\partial j})} w_{j\to j}(x_j) \prod_{k\in\partial j\setminus i} w_{k\to j}(x_k). \tag{B.15}$$

These equations, together with (B.10) and (B.11), are identical (apart from the notation) to the naive BP equations presented in [2]. From figure B.1 one sees that, even when the original graph is a tree, the corresponding factor graph contains short loops and the naive BP equations are not exact.

Current BP

We now consider an alternative form of the energy function (1) by introducing:

(i) a variable x_i^j for each vertex-neighbour pair $(i, j \in \partial i)$ (a kind of 'replica' of x_i);

(ii) a constraint imposing that all replicas of x_i are equal for each vertex i.

The constraints can be realized by assigning infinite energy penalties to configurations we want to be forbidden. Assuming $\gamma \to \infty$, we define

$$E(\{x_1^j\}_{j\in\partial 1}, \ldots, \{x_N^j\}_{j\in\partial N}) = \sum_{i=1}^{N} \left[\eta(x_i^*, \{x_j^i\}_{j\in\partial i}) + \gamma\, \chi(\{x_i^j\}_{j\in\partial i})\right], \tag{B.16}$$

where the function $\chi(\cdot)$ returns 1 when its entries are not all equal and 0 otherwise, whereas x_i^* means that the replica index is irrelevant. Note that the allowed (finite energy) configurations can be directly mapped onto the configurations of the original system, and also have the same Boltzmann weights. This does not depend on the specific form of the cluster energy function $\eta(\cdot)$, but only on the fact that each vertex of the original graph interacts (at most) with all its neighbours. With these definitions, each edge $\{i, j\}$ of the original graph can be naturally associated with the pair of variables $\{x_i^j, x_j^i\}$ (the j-replica of x_i and the i-replica of x_j). Moreover, the structure of the modified energy function (B.16) is described by a factor graph in which the variable nodes v correspond to the edges $\{i, j\}$ of the original graph, while the function nodes A now correspond to the clusters of interacting edges $A_i \equiv \{\{i, j\} | j \in \partial i\}$. Figure B.1 shows that, when the original graph is a tree, this factor graph is also one, and every variable node $\{i, j\}$ has degree 2, so that it only receives messages from the function nodes A_i and A_j. Using $m_{i\to j}$ as short-hand for $m_{A_i \to \{i,j\}}$, (B.2) becomes

$$p_{\{i,j\}}(x_i^j, x_j^i) = \mathrm{e}^{f_{\{i,j\}}}\, m_{i\to j}(x_i^j, x_j^i)\, m_{j\to i}(x_j^i, x_i^j). \tag{B.17}$$

The variable-to-function messages (B.4) are simply

$$w_{i\to j}(x_i^j, x_j^i) = m_{i\to j}(x_i^j, x_j^i), \tag{B.18}$$

where $w_{i\to j}$ is short-hand for $w_{\{i,j\}\to A_j}$. Finally, the cluster distribution (B.3) is

$$p_{A_i}(\{x_i^j, x_j^i\}_{j\in\partial i}) = \mathrm{e}^{f_{A_i} - \beta\eta(x_i^*, \{x_j^i\}_{j\in\partial i}) - \beta\gamma\chi(\{x_i^j\}_{j\in\partial i})} \prod_{j\in\partial i} m_{j\to i}(x_j^i, x_i^j), \tag{B.19}$$

where the cluster energy $\epsilon_A(x_A)$ has been replaced with the elementary term of (B.16). Discarding forbidden configurations (dropping replica indices), (B.17) is equivalent to (3) and, since all the χ-terms vanish, (B.19) is equivalent to (4). This is sufficient to derive the propagation equation (8), as shown in section 2. Finally, the free energy (B.7) is equivalent to (A.12), as all variable nodes of the factor graph have degree 2.

Appendix C. Simplified equations

In this appendix, we derive the simplified forms (9) and (10) of the propagation equation (8) and the free energy shift (6), respectively. Both derivations are based on similar manipulations. We consider the elementary energy term (2) associated with vertex i and note that it can be written in an alternative form for each given choice of a neighbour vertex $j \in \partial i$:

$$\eta(x_i, x_{\partial i}) = \sum_{x \in C \setminus x_i \setminus x_j} \prod_{k \in \partial i \setminus j} [1 - \delta(x_k, x)], \tag{C.1}$$

where the sum runs over the colour set C, excluding the colours x_i and x_j (if $x_i = x_j$, just one colour is excluded). Since the product in the equation above can only take the values 0 and 1, we can write the corresponding Boltzmann factor as

$$\mathrm{e}^{-\beta\eta(x_i, x_{\partial i})} = \prod_{x \in C \setminus x_i \setminus x_j} \left\{ 1 - (1 - \mathrm{e}^{-\beta}) \prod_{k \in \partial i \setminus j} [1 - \delta(x_k, x)] \right\}, \tag{C.2}$$

and expand the outer product

$$\mathrm{e}^{-\beta\eta(x_i, x_{\partial i})} = \sum_{B \subseteq C \setminus x_i \setminus x_j} (-1 + \mathrm{e}^{-\beta})^{|B|} \prod_{x \in B} \prod_{k \in \partial i \setminus j} [1 - \delta(x_k, x)], \tag{C.3}$$

where the sum runs over all the possible subsets B of the colour set $C \setminus x_i \setminus x_j$. Then, we exchange the two products, expand the product over x (taking into account that every product of two or more deltas vanishes), and use the fact that

$$\sum_{x \in C} \delta(x_k, x) = 1. \tag{C.4}$$

We finally obtain

$$\mathrm{e}^{-\beta\eta(x_i, x_{\partial i})} = \sum_{B \subseteq C \setminus x_i \setminus x_j} (-1 + \mathrm{e}^{-\beta})^{|B|} \prod_{k \in \partial i \setminus j} \sum_{x \in C \setminus B} \delta(x_k, x). \tag{C.5}$$

The propagation equation (8) for a given vertex i generates an outgoing message $m_{i \to j}(x_i, x_j)$ as a function of the set of incoming messages $m_{k \to i}(x_k, x_i)$ (where $k \in \partial i \setminus j$). Replacing the final expression for the Boltzmann factor (C.5) into this equation, we readily obtain the simplified propagation equation (9).

As far as the free energy shift (6) is concerned, we rewrite the elementary energy term (2) in yet another form, namely

$$\eta(x_i, x_{\partial i}) = \sum_{x \in C \setminus x_i} \prod_{j \in \partial i} [1 - \delta(x_j, x)]. \tag{C.6}$$

In this case the sum runs over the colour set C, excluding only the colour x_i. A totally analogous derivation allows us to write

$$\mathrm{e}^{-\beta\eta(x_i, x_{\partial i})} = \sum_{B \subseteq C \setminus x_i} (-1 + \mathrm{e}^{-\beta})^{|B|} \prod_{j \in \partial i} \sum_{x \in C \setminus B} \delta(x_j, x), \tag{C.7}$$

which plugged into (6), yields (10).

References

[1] Garey M and Johnson D S, 1979 *Computers and Intractability; A Guide to the Theory of NP-Completeness* (San Francisco, CA: Freeman)

[2] Bounkong S, van Mourik J and Saad D, 2006 *Phys. Rev.* E **74** 057101

[3] Wong K Y M and Saad D, 2008 *J. Phys. A: Math. Theor.* **41** 324023

[4] Yedidia J, 2001 *Advanced Mean Field Methods: Theory and Practice* ed M Opper and D Saad (Cambridge, MA: MIT Press) p 21
Yedidia J S, 2000 *Mitsubishi Electric Technical Report* 2000–27

[5] Yedidia J S, Freeman W T and Weiss Y, 2003 *Exploring Artificial Intelligence in the New Millennium* (San Francisco, CA: Morgan Kaufmann) p 239
Yedidia J S, Freeman W T and Weiss Y, 2001 *Mitsubishi Electric Technical Report* 2001-22

[6] Pearl J, 1986 *Artif. Intell.* **29** 241

[7] Pearl J, 1988 *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference* (San Francisco, CA: Morgan Kaufmann Publishers)

[8] Bethe H A, 1935 *Proc. R. Soc.* A **150** 552

[9] Burley D M, 1972 *Phase Transitions and Critical Phenomena* vol 2, ed C Domb and M S Green (New York: Academic) chapter 9

[10] Selman B, Kautz H A and Cohen B, 1994 *AAAI-94: Proc. of the 12th National Conf. on Artificial Intelligence* (Seattle, WA: MIT Press) p 337

[11] Biroli G and Mézard M, 2002 *Phys. Rev. Lett.* **88** 025501

[12] Pagnani A, Parisi G and Ratiéville M, 2003 *Phys. Rev.* E **67** 026116

[13] Yedidia J S, Freeman W T and Weiss Y, 2000 *NIPS: Advances in Neural Information Processing Systems* vol 13, p 689
Yedidia J S, Freeman W T and Weiss Y, 2000 *Mitsubishi Electric Technical Report* 2000–26

[14] Kikuchi R, 1951 *Phys. Rev.* **81** 988

[15] Pelizzola A, 2005 *J. Phys. A: Math. Gen.* **49** R309

[16] Mézard M and Parisi G, 2001 *Eur. Phys. J.* B **20** 217

[17] Mézard M, Parisi G and Virasoro M A, 1987 *Spin Glass Theory and Beyond* (Singapore: World Scientific)

[18] Mulet R, Pagnani A, Weigt M and Zecchina R, 2002 *Phys. Rev. Lett.* **89** 268701

[19] Krzakała F, Pagnani A and Weigt M, 2004 *Phys. Rev.* E **70** 046705

[20] Zdeborová L and Krzakała F, 2007 *Phys. Rev.* E **76** 031131

[21] Braunstein A and Zecchina R, 2006 *Phys. Rev. Lett.* **96** 030201

[22] Kschnischang F R, Frey B J and Loeliger H-A, 2001 *IEEE Trans. Inform. Theory* **47** 498