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

Loop series for discrete statistical models on graphs

Michael Chertkov[1] **and Vladimir Y Chernyak**[2]

[1] Theoretical Division and Center for Nonlinear Studies, LANL, Los Alamos, NM 87545, USA
[2] Department of Chemistry, Wayne State University, 5101 Cass Avenue, Detroit, MI 48202, USA
E-mail: chertkov@lanl.gov and chernyak@chem.wayne.edu

Received 3 April 2006
Accepted 17 May 2006
Published 14 June 2006

Online at stacks.iop.org/JSTAT/2006/P06009
doi:10.1088/1742-5468/2006/06/P06009

Abstract. In this paper we present the derivation details, logic, and motivation for the three loop calculus introduced in Chertkov and Chernyak (2006 *Phys. Rev.* E **73** 065102(R)). Generating functions for each of the three interrelated discrete statistical models are expressed in terms of a finite series. The first term in the series corresponds to the Bethe–Peierls belief–propagation (BP) contribution; the other terms are labelled by loops on the factor graph. All loop contributions are simple rational functions of spin correlation functions calculated within the BP approach. We discuss two alternative derivations of the loop series. One approach implements a set of local auxiliary integrations over continuous fields with the BP contribution corresponding to an integrand saddle-point value. The integrals are replaced by sums in the complementary approach, briefly explained in Chertkov and Chernyak (2006 *Phys. Rev.* E **73** 065102(R)). Local gauge symmetry transformations that clarify an important invariant feature of the BP solution are revealed in both approaches. The individual terms change under the gauge transformation while the partition function remains invariant. The requirement for all individual terms to be nonzero only for closed loops in the factor graph (as opposed to paths with loose ends) is equivalent to fixing the first term in the series to be exactly equal to the BP contribution. Further applications of the loop calculus to problems in statistical physics, computer and information sciences are discussed.

Keywords: rigorous results in statistical mechanics, analysis of algorithms, exact results, message-passing algorithms

ArXiv ePrint: cond-mat/0603189

Contents

1. Loop series for the factor graph and vertex models **5**
1.1. Bipartite factor graph model 5
1.2. Vertex models 6
1.2.1. Orientable vertex model. 7
1.2.2. General vertex model. 7
1.3. Loop series 8
1.3.1. General vertex model. 8
1.3.2. Orientable vertex model. 8
1.3.3. Factor graph model. 8

2. Loop series derivation for the vertex models **9**
2.1. Vertex model on orientable graphs 9
2.2. General vertex model 11

3. Loop series derivation for the factor graph model **13**
3.1. Integral representation for the factor graph model 13
3.2. Relation to the Bethe variational approach 14
3.3. Belief–propagation as a saddle-point 15
3.4. Gaussian correction to the saddle-point approximation 17
3.4.1. Loop expansion in the Gaussian case. 18
3.5. Loop calculus via integral representation 19

4. Conclusions **21**

Acknowledgments **23**

Appendix A: Bethe free energy **23**
Appendix A.1: Bethe free energy for the factor graph model 23
Appendix A.2: Bethe free energy for the general vertex model 25

Appendix B: Single loop example **26**

References **27**

One practically useful yet generally heuristic approach used for calculations of observables (correlation functions) in discrete statistical physics models, e.g. Ising model, is related to the so-called Bethe–Peierls (BP) approximation [1]–[3]. The BP approach is exact for graphs that do not contain loops, usually referred to as trees; otherwise the approach is approximate. The ad hoc approach can also be restated in a variational form [4]–[6]. A similar tree-based method in information science has been developed by Gallager [7, 8] in the context of error-correction theory. Gallager introduced the so-called low-density-parity-check (LDPC) codes, defined on locally tree-like Tanner graphs. The problem of ideal decoding, i.e., restoring the most probable pre-image out of the exponentially large pool of candidates, is identical to solving a statistical model on the graph [9]. An approximate yet efficient Belief–Propagation decoding algorithm introduced by Gallager constitutes an iterative solution of the Bethe–Peierls equations derived as if the statistical problem was defined on a tree that locally represents the Tanner graph. We utilize this abbreviation coincidence to call Bethe–Peierls and Belief–Propagation equations by the same acronym: BP. The recent resurgence of interest in LDPC codes [10, 11], as well as the proliferation of the BP approach to other areas of information and computer science, e.g. artificial intelligence [12] and combinatorial optimization [13]–[15], where interesting statistical models on graphs with long loops appear, has made the BP approach to be one of the most interesting and hot research topics in modern information and computer sciences.

In spite of the lack of analytical control in the general case of graphs/models with loops, the BP approximation and the corresponding algorithm provide remarkably accurate results. Based on this empirical observation one would expect an existence of a hidden mathematical structure that can rationalize an inessential, subleading role of the corrections associated with the loops. Besides, an in-depth understanding of the BP success would also provide a practical guidance for improving BP even further by accounting for nonlocal loop-related correlations. However, with the exception of two recent papers [16, 17], the discussion of this important point has been largely superficial and anecdotal. The Ising model (pairwise interactions between the bits) on a graph with loops has been considered by Montanari and Rizzo [16], where a set of exact equations has been derived that relates the correlation functions to each other. This system of equations is under-defined; however, if irreducible correlations are neglected the BP result is restored. This feature has been used [16] to generate a perturbative expansion for corrections to BP in terms of irreducible correlations. A complementary approach for the Ising model on a lattice has been taken by Parisi and Slanina [17], who utilized an integral representation developed by Efetov [18] in the early 1990s. The saddle-point for the integral representation used in [17] turns out to be exactly the BP solution. Calculating perturbative corrections to magnetization, the authors of [17] encountered divergences in their representation for the partition function; however, the divergences cancelled out from the leading-order correction to the magnetization revealing a sensible loop correction to BP. These papers, [16] and [17], became important initial steps towards calculating and understanding loop corrections to BP. However, both approaches are very far from being complete and problem-free. Thus, [16] lacks an invariant representation in terms of the partition function. Instead it requires operating with correlation functions. Besides, the complexity of the equations related to the higher-order corrections rapidly grows with the order. The complementary approach of [17] contains dangerous, since they lack analytical control, divergences (zero modes), which constitutes a very problematic symptom for any field theory. Both [16] and [17] focus on the Ising pairwise interaction model. The extensions of the proposed methods to the most interesting from the information theory viewpoint multi-bit interaction cases do not look straightforward. Finally, the approaches of [16] and [17], if extended to higher-order corrections, will result in infinite series. Resumming the corrections in all orders, so that the result is presented in terms of a finite series, does not look feasible within the proposed techniques.

In [19] we suggested an ultimate way to account for loop corrections to BP. We represented the partition function for a general discrete statistical model defined on a finite factor graph in terms of a series decomposition. The most remarkable feature of the suggested exact decomposition, that does not appear within the previous approaches [16, 17], is the representation of the partition function as a *finite* (!!) series with the first term being exactly represented by the BP solution. All higher-order terms are labelled by generalized loops in the factor graph. A generalized loop is defined as a possibly branching undirected path in the factor graph that has no loose ends. Each term in the series is represented as a product of local contributions along the loop, each contribution being expressed explicitly in terms of some correlation functions calculated within the BP approximation.

The present manuscript generalizes and details the approach of [19]. In addition to explaining all technical details of the loop series derivation of [19] we also provide an alternative approach based on an integral representation for the partition function. For the integral representation BP appears as a result of applying the saddle-point approximation. We pay special attention to clarifying the relation between the saddle-point approximation for the integral and the Bethe free energy approach of [6], as well as between the analysis of the Gaussian corrections and the saddle-point. We also provide a technical rationale for a formal gauge transformation in the integral representation for the model partition function (transformation of variables and decomposition of the integrand in a series) that results in the loop series expression.

The integral representation approach is formulated for a bipartite factor graph model [6, 20, 22] which is a particular case of the general vertex model of [21] also considered in the manuscript. For clarity of presentation we introduce a bipartite vertex model, an orientable vertex model, that is less general than the general vertex model, yet constitutes a generalization of the bipartite factor graph model[3]. The vertex models are more general compared to the bipartite factor graph model and allow a simpler derivation of the loop series using an auxiliary discrete transformation (discrete Fourier transforms) in place of its integral counterpart. We actually start the technical part of the paper by describing a simpler and more compact discrete variable representation before turning to a lengthy, still ideologically important, integral (continuous variables) counterpart.

The auxiliary degrees of freedoms, one per graph edge, introduced within both integral/continuous and sum/discrete approaches, possess a gauge symmetry that allows an invariant definition of the BP equation. Gauge transformation corresponding to the symmetry keeps the full expression for the partition function invariant while changing the individual terms of the series. An individual term corresponds to a path on the graph that may generally contain some number of loose ends. The BP equations can be viewed as conditions for fixing a special gauge that requires all allowed paths (i.e. those which contribute to the series) to be nothing but generalized loops that do not contain any loose ends. For this special gauge the BP approximation is described by the first bare term in the loop series.

The formulation of BP as a gauge fixing condition also allows a clear physical interpretation of the entire approach. Indeed, the first bare term of the loop series can be viewed as a 'ground state' that minimizes the Bethe free energy with loop corrections

[3] Examples of inference problems of information and computer sciences expressed in graphical terms can be found in [20]–[22]. Notice, however, that the simple modelling terminology we define and use in the manuscript is a bit different from the one used in information science. Thus, our general vertex model is equivalent to what is called the 'normal' factor graph model in [21] and 'Forney's factor graph' model in [22]. Our orientable graph model is equivalent to a normal factor graph which has been obtained by taking a standard bipartite factor graph of [20], and turning all the variable nodes into the so-called equality factor nodes [21].

being related to certain excited states described as along-the-loops spin flips with respect to the ground state. Such interpretation of the loop series makes our approach similar to the so-called high-temperature expansion, where individual contributions (diagrams) also correspond to close loops on the factor graph. There is, however, a very important key difference between the loop series and the high-temperature expansion. While the high-temperature expansion starts with a trivial bare term (just unity in the expansion of the partition function) the bare term in the loop series is highly nontrivial. It is represented by the BP approximation that already accounts for some local correlations in the model.

The manuscript is organized as follows. In section 1 we introduce our three basic models: bipartite factor graph model, orientable (bipartite) vertex, and general vertex models. Vertex models are convenient generalizations of the bipartite factor graph model. In section 1 we also state our major result—exact expressions for the models' partition functions in terms of finite series, coined loop series, over closed paths defined in the models' graphs. The rest of the paper is devoted to derivations and discussions of these results. A straightforward and simple derivation of the loop series for the vertex model is described in section 2. BP equations emerge as a result of a requirement for the finite series representation for the partition function of the model to have the loop series form (with no terms, correspondent to a path with loose ends, present). In section 3 we derive the loop series for the factor graph model via an integral representation. This derivation is more involved; however, we present it here in full as it allows us to establish a relation between the loop calculus and other approaches in theoretical physics, e.g. saddle-point analysis. Section 3 contains a number of subsections. Integral representation for the partition function of the factor node model is introduced in section 3.1. Section 3.2 describes the relation of the integral representation to the Bethe free energy variational approach of [6]. The latter is also briefly sketched in appendix A. In section 3.3 we present an approximate saddle-point analysis of the integral representation for the partition function of the factor graph model. Here we show that the saddle-point is described by the BP equations. The Gaussian approximation around the BP saddle-point is discussed in section 3.4. Finally, the derivation of the exact loop series via the integral representation is described in section 3.5. Section 4 is devoted to conclusions where we also discuss possible generalizations as well as practical utility of the loop series/calculus in information/computer science and statistical physics. Appendix B illustrates the loop calculus on a simple example of a two bit, two check bipartite factor graph model with single loop.

1. Loop series for the factor graph and vertex models

1.1. Bipartite factor graph model

Consider a generic discrete statistical model, with configurations characterized by a set of binary variables, $\sigma_i = \pm 1$, $i = 1, \ldots, n$, which is factorized so that the probability $p\{\sigma_i\}$ to find the system in the state $\{\sigma_i\}$ and the partition function Z are

$$p\{\sigma_i\} = Z^{-1} \prod_{\alpha} f_\alpha(\sigma_\alpha), \qquad Z = \sum_{\{\sigma_i\}} \prod_{\alpha} f_\alpha(\sigma_\alpha), \tag{1}$$

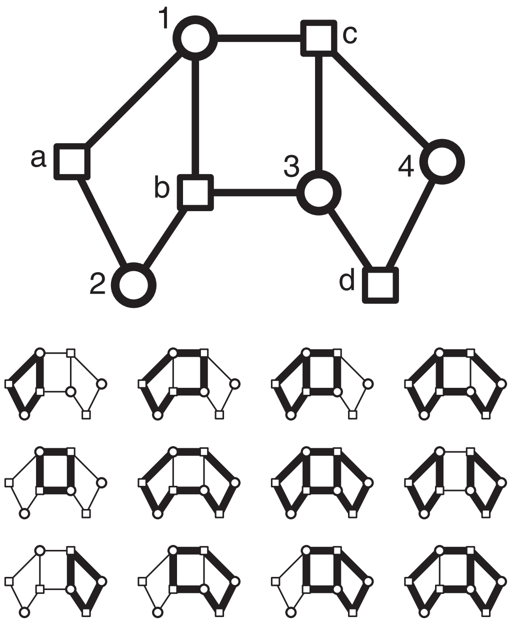

Figure 1. Example of a factor graph. Twelve possible marked paths (generalized loops) for the example are shown in bold on the bottom.

where α labels nonnegative and finite factor functions f_α with $\alpha = 1, \ldots, m$ and σ_α represents a subset of σ_i variables. Relations between factor functions (checks) and elementary discrete variables (bits), expressed as $i \in \alpha$ and $\alpha \ni i$, can be conveniently represented in terms of the system-specific factor graph. If $i \in \alpha$ we say that the bit and the check are neighbours. An example of a factor graph with $m = 4$ that corresponds to $p(\sigma_1, \sigma_2, \sigma_3, \sigma_4) = Z^{-1} f_a(\boldsymbol{\sigma}_a) f_b(\boldsymbol{\sigma}_b) f_c(\boldsymbol{\sigma}_c) f_d(\boldsymbol{\sigma}_d)$, where $\sigma_a \equiv (\sigma_1, \sigma_2)$, $\sigma_b \equiv (\sigma_1, \sigma_2, \sigma_3)$, $\sigma_c \equiv (\sigma_1, \sigma_3, \sigma_4)$, $\sigma_d \equiv (\sigma_3, \sigma_4)$ and $\alpha = a, b, c, d$, is shown in figure 1. Any spin correlation function can be calculated using the partition function, Z, defined by equation (1). For example, the bit i magnetization is expressed as

$$\langle \sigma_i \rangle = \left. \frac{\partial \ln Z}{\partial h_i} \right|_{h \to 0}, \qquad (2)$$

where the following transformation of a factor node function associated with a check α neighbouring bit i is assumed: $f_\alpha(\sigma_\alpha) \to f_\alpha(\sigma_\alpha) \exp(h_i \sigma_i)$.

1.2. Vertex models

In this section we discuss vertex models of two types, orientable/bipartite and general. Similar to the factor graph model, the vertex models are formulated in terms of Ising spin variables, $\sigma = \pm$. However, while in the factor graph model spins reside in the bit nodes, spins in the vertex models are assigned to the edges. The orientable vertex model generalizes the factor graph model described in section 1.1, while the general vertex model generalizes the orientable vertex model.

1.2.1. Orientable vertex model. A graph is orientable if the whole family of its nodes, X, can be partitioned in two subfamilies, such that nodes of one subfamily neighbours only nodes from the opposite subfamily. Also if a connected graph is orientable, there are exactly two different global orientations: a global orientation is chosen by picking some node on a graph and identifying it as left (or right). Choosing an orientation on an orientable graph partitions the set of nodes $X = X_\mathrm{L} \cup X_\mathrm{R}$ into the subsets of left and right nodes, referred to as bit nodes and check nodes, respectively. Ising variables in the vertex model reside in the graph edges, i.e., the configurations are defined by sets of Ising variables $\sigma_c = \pm 1$ for $c \in X_1$. For a graph with a chosen orientation it is also convenient to represent these variables as $\sigma_{j\alpha}$, with $\alpha \in X_\mathrm{R}$ and $j \in X_\mathrm{L}$ representing the check (right) and bit (left) end of an edge. The weight (probability) of a configuration is given by a product of weights related to the nodes:

$$p_\mathrm{ov}(\boldsymbol{\sigma}) = Z_\mathrm{ov}^{-1} \prod_{j \in X_\mathrm{L}} f_j\left(\boldsymbol{\sigma}_j\right) \prod_{\alpha \in X_\mathrm{R}} f_\alpha\left(\boldsymbol{\sigma}_\alpha\right), \qquad Z_\mathrm{ov} = \sum_{\{\boldsymbol{\sigma}\}} \prod_{j \in X_\mathrm{L}} f_j\left(\boldsymbol{\sigma}_j\right) \prod_{\alpha \in X_\mathrm{R}} f_\alpha\left(\boldsymbol{\sigma}_\alpha\right). \tag{3}$$

A particular example of the oriented vertex model defined for the graph shown in figure 1 corresponds to $p \sim f_1(\sigma_{1a}, \sigma_{1b}, \sigma_{1c}) f_2(\sigma_{2a}, \sigma_{2b}) f_3(\sigma_{3b}, \sigma_{3c}, \sigma_{3d}) f_4(\sigma_{4c}, \sigma_{4d})$ $f_a(\sigma_{a1}, \sigma_{a2}) f_b(\sigma_{b1}, \sigma_{b2}, \sigma_{b3}) f_c(\sigma_{c1}, \sigma_{c3}, \sigma_{c4}) f_d(\sigma_{d3}, \sigma_{d4})$, where we do not differentiate between the bits and checks, and the index order for a spin defined on the graph edge is not important.

Obviously, the oriented vertex model (3) turns into the factor graph model (1) if the functions f_j adopt the following form:

$$f_j(\boldsymbol{\sigma}_j) = \begin{cases} 1, & \sigma_{i\alpha} = \sigma_{i\beta} \quad \forall \alpha, \quad \beta \ni i \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

1.2.2. General vertex model. A general vertex model is determined by the weight function that can be represented in the following form:

$$p_\mathrm{gv}(\boldsymbol{\sigma}) = Z_\mathrm{gv}^{-1} \prod_{a \in X_0} f_a(\boldsymbol{\sigma}_a), \qquad Z_\mathrm{gv} = \sum_{\boldsymbol{\sigma}} \prod_{a \in X_0} f_a(\boldsymbol{\sigma}_a), \tag{5}$$

where a denotes a vertex in the model; elementary spin is defined at the edge connecting two neighbouring vertices, σ_{ab} for $b \in a$ and $a \in b$; $\boldsymbol{\sigma}_a$ stands for the vector built from all σ_{ab} where $b \in a$; $\boldsymbol{\sigma}$ is a particular configuration of spins on all the edges. It is important to realize that with this notation we need to assume that $\sigma_{ab} = \sigma_{ba}$.

A general vertex model turns into the orientable vertex model if the whole family of bits $\{a\}$ is divided in two subfamilies that correspond to checks and bits, $a = i \oplus \alpha$, and additionally for any bit/check the neighbours belong to the opposite families.

Therefore, for the example shown in figure 1 the oriented vertex model and the general vertex model simply coincide as the graph allows partitioning in two parts. A simple example of a general vertex model which does not correspond to any oriented case (the whole family of nodes is not divisible into two groups) is given by an interconnected triad of vertices with pair interaction: $p \sim f_1(\sigma_{12}, \sigma_{13}) f_2(\sigma_{21}, \sigma_{23}) f_3(\sigma_{31}, \sigma_{32})$.

1.3. Loop series

In this section we state the main result of the paper for the three models introduced above.

1.3.1. General vertex model. We start with the general vertex model. The partition function of the general vertex model, described by equation (5), is exactly equal to

$$Z_{\rm gv} = Z_0\left(1+\sum_C r_{\rm gv}(C)\right), \qquad r_{\rm gv}(C) = \frac{\prod_{a\in C}\mu_a}{\prod_{(ab)\in C}(1-m_{ab}^2)} \tag{6}$$

where the summation goes over all allowed C (marked) paths in the graph associated with the model; (ab) marks the edge on the graph connecting nodes a and b. The marked path is allowed to branch at any node/vertex; however, it cannot terminate at a node. We refer to such a structure as a loop (it is actually some kind of a generalized loop since branching is allowed; we use the shorter name for convenience). m_{ab} is the magnetization at the edge that connects nodes a and b. μ_a is the irreducible correlation function at node a. The order of the correlation function is equal to the number of marked nodes (nodes belonging to the marked path C) neighbouring a. The bare partition function Z_0, the magnetization m_{ab}, and the correlation functions μ_a are calculated within the BP procedure, described by equations (33) and (39)–(42).

1.3.2. Orientable vertex model. The general formula (6) applied to the case of the nodes/vertices partitioned into bits and checks, reads

$$Z_{\rm ov} = Z_0\left(1+\sum_C r_{\rm ov}(C)\right), \qquad r_{\rm ov}(C) = \frac{\left(\prod_{i\in C}\mu_i\right)\left(\prod_{\beta\in C}\mu_\beta\right)}{\prod_{(i\alpha)\in C}(1-m_{i\alpha}^2)} \tag{7}$$

where the summation goes over all allowed (marked) paths C in the graph associated with the model. A marked path (generalized loop) is allowed to branch at any bit/check; however, it may not terminate at a bit or check. In this case there are two types of irreducible correlation function associated with bits and checks, respectively, and one type of magnetization (which is associated with the edge that connects any bit with its neighbouring check that necessarily belongs to the loop) entering equation (7), all calculated within BP and defined in equations (29)–(31). Equation (7) also follows directly from the formulae of section 2.1.

1.3.3. Factor graph model. The decomposition of the partition function defined by equation (1) into a finite series has a form:

$$Z_{\rm fg} = Z_0\left(1+\sum_C r_{\rm fg}(C)\right), \qquad r_{\rm fg}(C) = \prod_{i,\alpha\in C}\mu_\alpha\mu_i, \tag{8}$$

$$\mu_i = \frac{(1-m_i)^{q_i-1}+(-1)^{q_i}(1+m_i)^{q_i-1}}{2(1-m_i^2)^{q_i-1}}, \qquad q_i = \sum_{\alpha\in C}^{\alpha\ni i} 1, \tag{9}$$

$$\mu_\alpha = \sum_{\sigma_\alpha} b_\alpha(\sigma_\alpha)\prod_{i\in C}^{i\in\alpha}(\sigma_i-m_i), \qquad m_i = \sum_{\sigma_i} b_i(\sigma_i)\sigma_i \tag{10}$$

where the summation goes over all allowed (marked) paths C (generalized loops). They consist of sets of bits and checks so that each of them has at least two distinct neighbours on the path. For the aforementioned example there are twelve allowed marked paths (loops) shown in figure 1 on the right. In equations (8) $b_i(\sigma_i)$, $b_\alpha(\sigma_\alpha)$ and Z_0 are beliefs (probabilities) defined on bits and checks and partition function, respectively, calculated for the BP solution. The BP solution for the model is described in detail in section 3; see also appendix A.

It is easy to verify that if equation (4) is assumed for the generalized vertex model, equation (7) turns exactly into equation (8). Indeed, under the condition of (4) the irreducible correlation functions at a check in the two formulae are exactly equivalent. One derives

$$\mu_i \to \int \mathrm{d}\sigma_i p_i(\sigma_i)(\sigma_i - m_i)^{q_i} = \frac{1-m_i^2}{2}\left[(1-m_i)^{q_i-1} + (-1)^{q_i}(1+m_i)^{q_i-1}\right], \tag{11}$$

$$\prod_{\alpha\in i,C}^{i\in C} (1-m_{i\alpha}^2) \to (1-m_i^2)^{q_i}, \tag{12}$$

where the definition of the $\mathrm{d}\sigma$ integration (summation) is given in section 2.1, $p_i(\sigma_i) = (1+\sigma_i m_i)/2$ is the probability to find bit i in the state σ_i within the BP solution, and q_i is the connectivity degree of bit i at the marked subgraph C, defined by equation (9). All together the equivalence is completely restored.

2. Loop series derivation for the vertex models

2.1. Vertex model on orientable graphs

To introduce a representation that leads to the loop expansion it is convenient to introduce simple integral calculus and discrete Fourier transform for functions $f(\sigma)$ of an Ising (spin) variable. Note that 'integrals' here are nothing but sums over discrete sets, introduced solely to simplify notations. A Fourier transform of $f(\sigma)$ is a function $\hat{\mathcal{F}}f(\pi)$, where the corresponding momentum $\pi = \pm 1$ is also an Ising variable. The definitions and properties of integrals and Fourier transform are as follows:

$$f(\sigma) = a + b\sigma; \qquad \int \mathrm{d}\sigma f(\sigma) = \sum_{\sigma=\pm1} f(\sigma); \qquad \int \mathrm{d}\sigma = 2; \qquad \int \sigma\, \mathrm{d}\sigma = 0; \tag{13}$$

$$\hat{\mathcal{F}}f(\pi) = \tfrac{1}{4}\int \mathrm{d}\sigma (1+\pi\sigma) f(\sigma); \qquad \hat{\mathcal{F}}^{-1}g(\sigma) = \int \mathrm{d}\pi (1+\pi\sigma) g(\pi);$$
$$\hat{\mathcal{F}}(1) = \frac{1}{2}; \qquad \hat{\mathcal{F}}(\sigma) = \frac{\pi}{2}; \qquad \hat{\mathcal{F}}^{-1}(1) = 2; \qquad \hat{\mathcal{F}}^{-1}(\pi) = 2\sigma. \tag{14}$$

doi:10.1088/1742-5468/2006/06/P06009

Denoting

$$F_\alpha(\boldsymbol{\sigma}_\alpha)=f_\alpha(\boldsymbol{\sigma}_\alpha);\qquad F_j(\boldsymbol{\pi}_j)=\hat{\mathcal{F}}f_j(\boldsymbol{\pi}_j);\qquad f_j(\boldsymbol{\sigma})=\int \mathrm{d}\boldsymbol{\pi}_j F_j(\boldsymbol{\pi}_j)\prod_{\alpha\ni j}(1+\pi_{j\alpha}\sigma_{\alpha j}), \tag{15}$$

where $\mathrm{d}\boldsymbol{\pi}_j=\prod_{\alpha\ni j}\mathrm{d}\pi_{j\alpha}$, we can represent the partition function in the form

$$Z_{\mathrm{ov}}=\int\prod_\alpha \mathrm{d}\boldsymbol{\sigma}_\alpha\,\mathrm{d}\boldsymbol{\pi}_\alpha\left(\prod_j F_j(\boldsymbol{\pi}_j)\right)\left(\prod_\alpha F_\alpha(\boldsymbol{\sigma}_\alpha)\right)\left(\prod_{j\alpha}(1+\pi_{j\alpha}\sigma_{\alpha j})\right). \tag{16}$$

Here and below in this subsection the index order in the definition of the discrete fields is arbitrary, i.e., $\sigma_{i\alpha}=\sigma_{\alpha i}$. Our derivation of the loop expansion rests on an important, yet very simple relation that can be easily verified directly:

$$\frac{\cosh(\eta+\chi)(1+\pi\sigma)}{(\cosh\eta+\sigma\sinh\eta)(\cosh\chi+\pi\sinh\chi)}\\ =1+(\tanh(\eta+\chi)-\sigma)\,(\tanh(\eta+\chi)-\pi)\cosh^2(\eta+\chi). \tag{17}$$

Introducing two sets of parameters $\eta_{\alpha j}$ and $\chi_{j\alpha}$ that reside in the graph edges we can make use of equation (17) to re-group the terms. This results in the following expression for the partition function:

$$Z_{\mathrm{ov}}=\bar{Z}_{\mathrm{gv}}\int \mathrm{d}\boldsymbol{\sigma}\,\mathrm{d}\boldsymbol{\pi}\prod_j P_j(\boldsymbol{\pi}_j)\prod_\alpha P_\alpha(\boldsymbol{\sigma}_\alpha)\prod_{j\alpha}V_{j\alpha}\left(\sigma_{\alpha j},\pi_{j\alpha}\right), \tag{18}$$

$$\bar{Z}_{\mathrm{gv}}=\left(\prod_{j\alpha}\cosh\left(\eta_{\alpha j}+\chi_{j\alpha}\right)\right)^{-1}, \tag{19}$$

$$P_j(\boldsymbol{\pi}_j)=F_j(\boldsymbol{\pi}_j)\prod_{\alpha\ni j}\left(\cosh(\chi_{j\alpha})+\pi_{j\alpha}\sinh(\chi_{j\alpha})\right), \tag{20}$$

$$P_\alpha(\boldsymbol{\sigma}_\alpha)=F_\alpha(\boldsymbol{\sigma}_j)\prod_{j\in\alpha}\left(\cosh(\eta_{\alpha j})+\sigma_{\alpha j}\sinh(\eta_{\alpha j})\right), \tag{21}$$

$$V_{j\alpha}\left(\sigma_{\alpha j},\pi_{j\alpha}\right)=1+\left(\tanh(\eta_{\alpha j}+\chi_{j\alpha})-\sigma_{\alpha j}\right)\left(\tanh(\eta_{\alpha j}+\chi_{j\alpha})-\pi_{j\alpha}\right)\cosh^2(\eta_{\alpha j}+\chi_{j\alpha}). \tag{22}$$

The desired decomposition is obtained by expanding the V-terms followed by a local computation. The parameters $\boldsymbol{\eta}$ and $\boldsymbol{\chi}$ are chosen using the criterion that skeletons (subgraphs) with loose ends do not contribute to the decomposition. This can be achieved if the parameters satisfy the following system of equations:

$$\int \mathrm{d}\boldsymbol{\pi}_j\left(\tanh(\eta_{\alpha j}+\chi_{j\alpha})-\pi_{j\alpha}\right)P_j(\boldsymbol{\pi}_j)=0, \tag{23}$$

$$\int \mathrm{d}\boldsymbol{\sigma}_\alpha\left(\tanh(\eta_{\alpha j}+\chi_{j\alpha})-\sigma_{\alpha j}\right)P_\alpha(\boldsymbol{\sigma}_\alpha)=0. \tag{24}$$

The first equation in the system, equation (23), can actually be reduced by making use of equations (13)–(15) and (20), to

$$\int \mathrm{d}\boldsymbol{\sigma}_j\left(\tanh(\eta_{\alpha j}+\chi_{j\alpha})-\sigma_{j\alpha}\right)\tilde{P}_j(\boldsymbol{\sigma}_j)=0, \tag{25}$$

$$\tilde{P}_j(\boldsymbol{\sigma}_j)=f_j(\boldsymbol{\sigma}_j)\prod_{\alpha\ni j}\left(\cosh(\chi_{j\alpha})+\sigma_{j\alpha}\sinh(\chi_{j\alpha})\right). \tag{26}$$

Combining equations (24), (25) we derive

$$\frac{\exp\left[(\eta_{\alpha j}+\chi_{j\alpha})\sigma_{j\alpha}\right]}{\cosh\left[\eta_{\alpha j}+\chi_{j\alpha}\right]}=\sum_{\boldsymbol{\sigma}_j\backslash\sigma_{j\alpha}}b_j^{(\mathrm{ov})}=\sum_{\boldsymbol{\sigma}_\alpha\backslash\sigma_{\alpha j}}b_\alpha^{(\mathrm{ov})}, \tag{27}$$

$$b_j^{(\mathrm{ov})}=\frac{\tilde{P}_j(\boldsymbol{\sigma}_j)}{\sum_{\boldsymbol{\sigma}_j}\tilde{P}_j(\boldsymbol{\sigma}_j)},\qquad b_\alpha^{(\mathrm{ov})}=\frac{P_\alpha(\boldsymbol{\sigma}_\alpha)}{\sum_{\boldsymbol{\sigma}_\alpha}P_\alpha(\boldsymbol{\sigma}_\alpha)}, \tag{28}$$

where it is assumed that $\sigma_{j\alpha}=\sigma_{\alpha j}$. Equation (27) constitutes the BP system of equations, represented in terms of parameters η and χ. Equations (28) provide the BP expressions for the probabilities (beliefs) to observe the spin vector associated with a bit/check in the corresponding states.

A typical sum/integral, needed to calculate individual marked path/diagram C contribution, is reduced to the following irreducible correlation functions that should be computed within BP:

$$\mu_\alpha=\int \mathrm{d}\boldsymbol{\sigma}_\alpha b_\alpha^{(\mathrm{ov})}(\boldsymbol{\sigma}_\alpha)\prod_{i\in\alpha,C}\left(m_{i\alpha}-\sigma_{i\alpha}\right), \tag{29}$$

$$\mu_i=\int \mathrm{d}\boldsymbol{\sigma}_i\, b_i^{(\mathrm{gv})}(\boldsymbol{\sigma}_i)\prod_{\alpha\ni i;\alpha\in C}\left(m_{i\alpha}-\sigma_{i\alpha}\right), \tag{30}$$

where $m_{i\alpha}$ is the BP magnetization at the edge $i\alpha$:

$$m_{i\alpha}=\int \mathrm{d}\boldsymbol{\sigma}_i b_i^{(\mathrm{ov})}(\boldsymbol{\sigma}_i)\sigma_{i\alpha}=\int \mathrm{d}\boldsymbol{\sigma}_\alpha b_\alpha^{(\mathrm{ov})}(\boldsymbol{\sigma}_\alpha)\sigma_{i\alpha}. \tag{31}$$

2.2. General vertex model

We are now in a position to consider the case of a general, not necessarily orientable, graph. The loop expansion and the BP equations can be readily extended to this case. To derive the desired loop decomposition we relax the condition $\sigma_{ab}=\sigma_{ba}$, i.e., we treat σ_{ab} and σ_{ba} as independent Ising variables. In complete analogy with the orientable case we represent the partition function in a form

$$Z_{\mathrm{gv}}=\int \mathrm{d}\boldsymbol{\sigma}\prod_a f_a(\boldsymbol{\sigma}_a)\prod_{bc}\frac{1+\sigma_{bc}\sigma_{cb}}{2}. \tag{32}$$

Note that for this representations the vectors $\boldsymbol{\sigma}_a$ become independent variables. Also in the product over bc we assume that each edge contributes only once. We further introduce

doi:10.1088/1742-5468/2006/06/P06009

a parameter vector $\boldsymbol{\eta}$ with the components η_{ab}, all of them being independent variables. Making use of equation (17) we arrive at the following representation for the partition function that is ready for the loop decomposition:

$$Z_{\mathrm{gv}} = \bar{Z}_{\mathrm{gv}} \int \mathrm{d}\boldsymbol{\sigma} \prod_a P_a(\boldsymbol{\sigma}_a) \prod_{bc} V_{bc}\left(\sigma_{bc}, \sigma_{cb}\right); \qquad \bar{Z}_{\mathrm{gv}} = \left(\prod_{bc} 2\cosh\left(\eta_{bc}+\eta_{cb}\right)\right)^{-1};$$

$$P_a(\boldsymbol{\sigma}_a) = f_a(\boldsymbol{\sigma}_a) \prod_{b\in a} \left(\cosh\eta_{ab} + \sigma_{ab}\sinh\eta_{ab}\right); \tag{33}$$

$$V_{bc}\left(\sigma_{bc}, \sigma_{cb}\right) = 1 + \left(\tanh(\eta_{bc}+\eta_{cb}) - \sigma_{bc}\right)\left(\tanh(\eta_{bc}+\eta_{cb}) - \sigma_{cb}\right)\cosh^2(\eta_{bc}+\eta_{cb}). \tag{34}$$

The BP equations for our general case have a form

$$\int \mathrm{d}\boldsymbol{\sigma}_a \left(\tanh(\eta_{ab}+\eta_{ba}) - \sigma_{ab}\right) P_a(\boldsymbol{\sigma}_a) = 0. \tag{35}$$

To recast equation (35) in a standard BP form we denote by $\boldsymbol{\eta}_{ab}$ the vector with the components η_{ac} with $c \in a$ and $c \neq b$, i.e., $\boldsymbol{\eta}_a = (\boldsymbol{\eta}_{ab}, \eta_{ab})$. We also define a function $\gamma(\boldsymbol{\eta}_{ab})$ using the condition

$$\int \prod_{c\in a}^{c\neq b} \mathrm{d}\sigma_{ac}\, f_a(\boldsymbol{\sigma}_a) \prod_{c\in a}^{c\neq b} \left(\cosh\eta_{ac} + \sigma_{ac}\sinh\eta_{ac}\right) = \phi(\cosh\gamma + \sigma_{ab}\sinh\gamma). \tag{36}$$

The meaning of equation (36) is as follows. The lhs of the equation is a function of the Ising variable σ_{ab} and a function of $\boldsymbol{\eta}_{ab}$ (since by definition it does not depend on η_{ab}). The rhs constitutes a generic representation of such a function provided ϕ and γ are allowed to depend on $\boldsymbol{\eta}_{ab}$. Integrating (summing) over σ_{ab} in equation (36) with and without the σ_{ab} factor allows us to determine the function $\gamma(\boldsymbol{\eta}_{ba})$ explicitly:

$$\tanh\gamma(\boldsymbol{\eta}_{ab}) = \frac{\int \mathrm{d}\boldsymbol{\sigma}_a\, \sigma_{ab} f_a(\boldsymbol{\sigma}_a) \prod_{c\in a}^{c\neq b}\left(\cosh\eta_{ac} + \sigma_{ac}\sinh\eta_{ac}\right)}{\int \mathrm{d}\boldsymbol{\sigma}_a f_a(\boldsymbol{\sigma}_a) \prod_{c\in a}^{c\neq b}\left(\cosh\eta_{ac} + \sigma_{ac}\sinh\eta_{ac}\right)}. \tag{37}$$

Multiplying equation (36) with a factor $(\cosh\eta_{ab} + \sigma_{ab}\sinh\eta_{ab})$ yields

$$\int \prod_{c\in a}^{b\neq b} \mathrm{d}\sigma_{ac} P_a(\boldsymbol{\sigma}_a)\phi\left(\cosh(\gamma+\eta_{ab}) + \sigma_{ab}\sinh(\gamma+\eta_{ab})\right). \tag{38}$$

Comparing equation (38) with equation (35), we arrive at $\sinh(\gamma - \eta_{ba}) = 0$. This allows us to represent the BP equations in a more conventional form:

$$\eta_{ba} = \gamma\left(\boldsymbol{\eta}_{ab}\right). \tag{39}$$

Calculated within BP, the probability of finding the whole family of edges connected to node a in the state $\boldsymbol{\sigma}_a$ is

$$b_a^{(\mathrm{gv})}(\boldsymbol{\sigma}_a) = \frac{P_a(\boldsymbol{\sigma}_a)}{\int \mathrm{d}\boldsymbol{\sigma}_a\, P_a(\boldsymbol{\sigma}_a)}. \tag{40}$$

In the general vertex model case a typical integral (sum), needed to take to calculate a diagram contribution for a generalized loop C, is reduced to the corresponding irreducible correlation functions of the spin variables computed within BP:

$$\mu_a = \int \mathrm{d}\boldsymbol{\sigma}_a \, b_a^{(\mathrm{gv})}(\boldsymbol{\sigma}_a) \prod_{b\in a,C} (m_{ab} - \sigma_{ab}) \,, \tag{41}$$

where m_{ab} is the magnetization at the edge (ab) calculated within BP:

$$m_{ab} = \int \mathrm{d}\boldsymbol{\sigma}_a \, b_a^{(\mathrm{gv})}(\boldsymbol{\sigma}_a)\sigma_{ab}. \tag{42}$$

The final and most general expression equation (6) emerges in the result of direct calculation of the (generalized) loop contributions making use of equations (34), (39), (41), (42).

3. Loop series derivation for the factor graph model

3.1. Integral representation for the factor graph model

We aim to derive a convenient integral representation for the statistical model (1). As a first step we introduce two statistically independent sets of discrete random variables: the original $\{\sigma_i\}$, and the additional factor-variable counterpart $\{\pi_\alpha\}$, where each π_α is a vector consisting of q_α scalar components, each a discrete random variable, and q_α is the degree of connectivity of the corresponding factor node. For the example represented by figure (1) we have $\pi_a = (\pi_a^{(1)}, \pi_a^{(2)}), \pi_b = (\pi_b^{(1)}, \pi_b^{(2)}), \pi_c = (\pi_c^{(1)}, \pi_c^{(3)}, \pi_c^{(4)}), \pi_d = (\pi_c^{(3)}, \pi_c^{(4)})$ where $\pi_{a,b,c,d}^{(i)} = \pm 1$. Using such a representation the partition function of equation (1) can be rewritten as

$$Z \sim \sum_{\{\pi_\alpha\}} \left[\prod_\alpha f_\alpha(\pi_\alpha) \right] \prod_i \left[\sum_{\sigma_i} \prod_{\alpha\ni i} \delta\left(\sigma_i, \pi_\alpha^{(i)}\right) \right], \tag{43}$$

where the product over i is taken over the bits connected to more then one factor nodes. Under the condition that all discrete scalars $\pi_\alpha^{(i)}$ belong to the binary alphabet the expression on the rhs of equation (43) can be rewritten as

$$\sum_{\sigma_i} \prod_{\alpha\ni i} \delta\left(\sigma_i, \pi_\alpha^{(i)}\right) \sim \int_{\mathbf{C}_i} \mathrm{d}\boldsymbol{\chi}_i \exp\left(\sum_{\alpha\ni i} \chi_{i\alpha} \pi_\alpha^{(i)} \right) \left[\sum_{\sigma_i} \exp\left(\frac{1}{q_i - 1} \sigma_i \sum_{\alpha\ni i} \chi_{i\alpha} \right) \right]^{1-q_i} , \tag{44}$$

where $q_i > 1$ is the degree of connectivity of bit i, and $\boldsymbol{\chi}_i$ is a vector with the components $\chi_{i\alpha}$, where $\alpha \ni i$. Integration goes over a q_i-dimensional cycle $\mathbf{C}_i = \prod_{\alpha\ni i} C_{i\alpha}$ that constitutes a cartesian product of q_i contours in the complex plane: $C_{j\alpha}$ connects the points $z_{j\alpha}$ and $z_{j\alpha} + 2\pi i\,(q_j - 1)$ in an arbitrary way, however such that the contour does not go through the point of formal singularity of the integrand in equation (44). It is straightforward to check that the result does not depend on the particular choice of the reference points $z_{i\alpha}$. The multidimensional integration contour can be defined in way (in the sense of passing the multidimensional pole manifold) that the integral representation is exact, yet its deformation that reaches the saddle-point does not involve the pole manifold. This is confirmed indirectly by identical exact loop expansions that originate from the integral representation and its discrete counterpart. Note also that the integral representation is obviously not unique and the specific choice of the representation is dictated by our desire to find one that guarantees emergence of the BP in the saddle-point approximation applied to the integral. Below in section 3.3 we will verify, indeed, that equation (44) obeys the desired saddle-point property.

Substituting equation (44) into equation (43) one derives

$$\begin{aligned} Z &\sim \int \left[\prod_i\prod_\alpha \mathrm{d}\chi_{i\alpha}\right]\prod_i\left[\sum_{\sigma_i}\exp\left(\frac{1}{q_i-1}\,\sigma_i\sum_{\alpha\ni i}\chi_{i\alpha}\right)\right]^{1-q_i} \\ &\qquad \times \prod_\alpha\left[\sum_{\pi_\alpha}\left(f_\alpha(\pi_\alpha)\exp\left(\sum_{i\in\alpha}\pi_\alpha^{(i)}\chi_{i\alpha}\right)\right)\right] \qquad (45)\\ &= \int \left[\prod_i\prod_\alpha \mathrm{d}\chi_{i\alpha}\right]\left(\prod_i \exp\left[-Q_i(\boldsymbol{\chi})\right]\right)\left(\prod_\alpha\left[\sum_{\pi_\alpha}\exp\left[-Q_\alpha(\boldsymbol{\chi})\right]\right]\right) \\ &= \int \left[\prod_i\prod_\alpha \mathrm{d}\chi_{i\alpha}\right]\exp\left[-\mathcal{S}_0(\chi)\right]. \qquad (46)\end{aligned}$$

3.2. Relation to the Bethe variational approach

The expression of equation (45) is compact and already constitutes a good starting point for further, e.g. saddle-point, analysis. Meantime, for the purpose of establishing a relation to the Bethe free energy approach of [6] and for some further applications we introduce the following auxiliary integrations,

$$1 \sim \int\left[\prod_i\prod_{\sigma_i}\mathrm{d}\varphi_i(\sigma_i)\,\mathrm{d}\bar\varphi_i(\sigma_i)\right]\exp\left[\sum_i\sum_{\sigma_i}\bar\varphi_i(\sigma_i)\left(\varphi_i(\sigma_i)-\sigma_i\sum_{\alpha\ni i}\chi_{i\alpha}\right)\right], \qquad (47)$$

$$\begin{aligned} 1 &\sim \int\left[\prod_\alpha\prod_{\pi_\alpha}\mathrm{d}\psi_\alpha(\pi_\alpha)\,\mathrm{d}\bar\psi_\alpha(\pi_\alpha)\right] \\ &\qquad \times\exp\left[\sum_\alpha\sum_{\pi_\alpha}\bar\psi_\alpha(\pi_\alpha)\left(\psi_\alpha(\pi_\alpha)+\ln f_\alpha(\pi_\alpha)+\sum_{i\in\alpha}\pi_\alpha^{(i)}\chi_{i\alpha}\right)\right], \qquad (48)\end{aligned}$$

in the rhs of equation (45). After some obvious manipulations we arrive at

$$Z \sim \int\left[\prod_i\prod_\alpha\mathrm{d}\chi_{i\alpha}\right]\left[\prod_i\prod_{\sigma_i}\mathrm{d}\varphi_i(\sigma_i)\,\mathrm{d}\bar\varphi_i(\sigma_i)\right]\left[\prod_\alpha\prod_{\pi_\alpha}\mathrm{d}\psi_\alpha(\pi_\alpha)\mathrm{d}\bar\psi_\alpha(\pi_\alpha)\right]\exp\left[-\mathcal{S}\right], \qquad (49)$$

$$\begin{aligned}\mathcal{S} &= \sum_i\left[-\sum_{\sigma_i}\bar\varphi_i(\sigma_i)\left(\varphi_i(\sigma_i)-\sigma_i\sum_{\alpha\ni i}\chi_{i\alpha}\right)+(q_i-1)\ln\left(\sum_{\sigma_i}\exp\left(\frac{\varphi_i(\sigma_i)}{q_i-1}\right)\right)\right] \\ &\qquad -\sum_\alpha\left[\sum_{\pi_\alpha}\bar\psi_\alpha(\pi_\alpha)\left(\psi_\alpha(\pi_\alpha)+\ln f_\alpha(\pi_\alpha)+\sum_{i\in\alpha}\pi_\alpha^{(i)}\chi_{i\alpha}\right)\right. \\ &\qquad \left.+\ln\left(\sum_{\pi_\alpha}\exp\left(-\psi_\alpha(\pi_\alpha)\right)\right)\right]. \qquad (50)\end{aligned}$$

Evaluating the integral over $\varphi_i(\sigma_i)$, $\psi_\alpha(\pi_\alpha)$ within the saddle-point approximation we obtain

$$\bar{\varphi}_i(\sigma_i) = \frac{\exp\left(\varphi_i^{(\mathrm{sp})}(\sigma_i)/(q_i-1)\right)}{\sum_{\sigma_i}\exp\left(\varphi_i^{(\mathrm{sp})}(\sigma_i)/(q_i-1)\right)} = (z_i^{(\mathrm{sp})})^{-1}\exp\left(\varphi_i^{(\mathrm{sp})}(\sigma_i)/(q_i-1)\right), \tag{51}$$

$$\bar{\psi}_\alpha(\pi_\alpha) = \frac{\exp\left(-\psi_\alpha^{(\mathrm{sp})}(\pi_\alpha)\right)}{\sum_{\pi_\alpha}\exp\left(-\psi_\alpha^{(\mathrm{sp})}(\pi_\alpha)\right)} = (z_\alpha^{(\mathrm{sp})})^{-1}\exp\left(-\psi_\alpha^{(\mathrm{sp})}(\pi_\alpha)\right). \tag{52}$$

Expressing $\varphi_i(\sigma_i)$, $\psi_\alpha(\pi_\alpha)$ in terms of $\bar{\varphi}_i(\sigma_i)$, $\bar{\psi}_\alpha(\pi_\alpha)$ according to equations (51), (52) and substituting the result into the effective action (50), we find

$$\begin{aligned}\mathcal{S}^{(\mathrm{sp})} = &-\sum_\alpha\sum_{\pi_\alpha}\bar{\psi}_\alpha(\pi_\alpha)\ln f_\alpha(\pi_\alpha)\\ &+\sum_\alpha\sum_{\pi_\alpha}\bar{\psi}_\alpha(\pi_\alpha)\ln\bar{\psi}_\alpha(\pi_\alpha)-\sum_i\sum_{\sigma_i}(q_i-1)\bar{\varphi}_i(\sigma_i)\ln\bar{\varphi}_i(\sigma_i)\\ &+\sum_i\sum_{\alpha\ni i}\chi_{i\alpha}\sum_{\sigma_i}\sigma_i\left(\bar{\varphi}_i(\sigma_i)-\sum_{\pi_\alpha\backslash\sigma_i}\bar{\psi}_\alpha(\pi_\alpha)\right)\\ &-\sum_\alpha\ln z_\alpha^{(\mathrm{sp})}+\sum_i(q_i-1)\ln z_i^{(\mathrm{sp})}.\end{aligned} \tag{53}$$

The saddle-point (in ψ and φ) solution (51), (52) is highly degenerate: there is a freedom in imposing a constraint for any bit i and per any factor node α. Moreover, the integrand in equations (49) and (50) is invariant under the transformations:

$$\psi_\alpha(\pi_\alpha)\to\psi_\alpha(\pi_\alpha)+c_\alpha,\qquad \varphi_i(\sigma_i)\to\varphi_i(\sigma_i)+c_i. \tag{54}$$

Fixing the values of $\sum_{\sigma_i}\varphi_i(\sigma_i)$ and $\sum_{\pi_\alpha}\psi_\alpha(\pi_\alpha)$, introducing the shifts (54) into equations (49), (50) and integrating with respect to c_i, c_α, one arrives at the normalization constraints

$$\sum_{\sigma_i}\bar{\varphi}_i(\sigma_i)=1,\qquad \sum_{\pi_\alpha}\bar{\varphi}_i(\pi_\alpha)=1, \tag{55}$$

that are dynamically imposed, i.e., they are present in the integrand of equation (49) as products of the corresponding sets of δ-functions. A convenient choice of $\sum_{\sigma_i}\varphi_i(\sigma_i)$ and $\sum_{\pi_\alpha}\psi_\alpha(\pi_\alpha)$ constraints is the one that makes $z_i^{(\mathrm{sp})}=z_\alpha^{(\mathrm{sp})}=1$. As a result the last two terms on the rhs of equation (53) disappear and the equivalence between the effective action (53) and the Bethe free energy of [6] (see also appendix A) becomes clear.

3.3. Belief–propagation as a saddle-point

Looking for the saddle-point configurations of the auxiliary fields $\bar{\varphi}_i(\sigma_i)$, $\bar{\psi}_\alpha(\pi_\alpha)$, $\varphi_i(\sigma_i)$, $\psi_\alpha(\pi_\alpha)$ and $\chi_{i\alpha}$ that dominate the contribution to the integral in equation (49), and thus setting the corresponding partial derivatives of $\mathcal{S}$ to zero, we obtain in addition to

doi:10.1088/1742-5468/2006/06/P06009

equations (51), (52) the following saddle-point equations:

$$\varphi_i^{(\mathrm{sp})}(\sigma_i) = \sigma_i \sum_{\alpha\ni i} \chi_{i\alpha}^{(\mathrm{sp})}, \tag{56}$$

$$\psi_\alpha^{(\mathrm{sp})}(\pi_\alpha) + \ln f_\alpha(\pi_\alpha) + \sum_{i\in\alpha} \pi_\alpha^{(i)} \chi_{i\alpha}^{(\mathrm{sp})} = 0, \tag{57}$$

$$\sum_{\sigma_i} \sigma_i \bar{\varphi}_i^{(\mathrm{sp})}(\sigma_i) = \sum_{\pi_\alpha} \pi_\alpha^{(i)} \bar{\psi}_\alpha^{(\mathrm{sp})}(\pi_\alpha). \tag{58}$$

This system of equations (51), (52) and (56)–(58) is identical to the BP system of equation derived via variation of the Bethe free energy [6] (see also appendix A). The relation between the corresponding fields is as follows: $\bar{\varphi}_i^{(\mathrm{sp})}(\sigma_i) \leftrightarrow b_i(\sigma_i)$, $\bar{\psi}_\alpha^{(\mathrm{sp})}(\pi_\alpha) \leftrightarrow b_\alpha(\pi_\alpha)$, $\chi_{i\alpha}^{(\mathrm{sp})}\sigma_i \leftrightarrow \lambda_{i\alpha}(\sigma_i)$, $\chi_{i\alpha}^{(\mathrm{sp})} \leftrightarrow \sum_{\beta\ni i}^{\beta\neq\alpha} \eta_{i\beta}$. Normalization constraints (A.2) are obviously satisfied in equations (51) and (52). The consistency constraint (A.3) is equivalent to (58).

Equations (51), (52) and (56)–(58) result in

$$\frac{\sum_{\pi_\alpha\setminus\sigma_i} f_\alpha(\pi_\alpha)\exp\left(\sum_{j\in\alpha}\pi_\alpha^{(j)}\chi_{j\alpha}^{(\mathrm{sp})}\right)}{\sum_{\pi_\alpha} f_\alpha(\pi_\alpha)\exp\left(\sum_{j\in\alpha}\pi_\alpha^{(j)}\chi_{j\alpha}^{(\mathrm{sp})}\right)} = \frac{\exp\left(\sigma_i \sum_{\beta\ni i}\chi_{i\beta}^{(\mathrm{sp})}/(q_i-1)\right)}{\sum_{\sigma_i}\exp\left(\sigma_i \sum_{\beta\ni i}\chi_{i\beta}^{(\mathrm{sp})}/(q_i-1)\right)} \tag{59}$$

that can also be derived directly from equation (45). The lhs or rhs of equation (59) gives the saddle-point, BP expression for $\bar{\varphi}_i(\sigma_i)$—the probability to observe spin at bit i in the state σ_i.

The set of equations (51), (52), and (56)–(45) coincides with the one derived as an extremum condition for the Bethe free energy [6] (see also appendix A). Iterative solution of these nonlinear equations reproduces the famous Belief–Propagation algorithm for efficient yet suboptimal solution of the inference problem.

The saddle-point approximation for the partition function

$$Z_0 \sim \exp\left[-\mathcal{S}_0\left(\chi^{(\mathrm{sp})}\right)\right], \tag{60}$$

is expressed in terms of the effective action $\mathcal{S}_0$ defined in (45) and (46). Note that there may be more than one realizable (correspondent to real valued $\chi^{(\mathrm{sp})}$) solution of the saddle-point (BP) system of equations.

For the purpose of further applications let us also introduce the magnetization and irreducible correlation function (defined at two bits neighbouring the same check), both defined within the saddle-point BP approximation:

$$m_i = \sum_{\sigma_i} \sigma_i \bar{\varphi}_i^{(\mathrm{sp})}(\sigma_i), \tag{61}$$

$$\text{for } i,j\in\alpha: \quad \mu_{ij} = \sum_{\sigma_\alpha}(\sigma_i - m_i)(\sigma_j - m_j)\bar{\psi}_\alpha^{(\mathrm{sp})}(\sigma_\alpha). \tag{62}$$

3.4. Gaussian correction to the saddle-point approximation

To calculate the correction to the zero-order saddle-point approximation we need to expand the effective action $\mathcal{S}_0$ in $\chi_{i\alpha}$ around $\chi^{(\mathrm{sp})}$ to the second order. According to the definition of the saddle-point the first-order term in the expansion is exactly zero. If the second-order expansion is sufficient, i.e., the higher-order terms are much smaller with respect to some parameter (the exact origin of the expansion parameter will be verified and discussed later), we shift the multidimensional integration contour that enters equation (46) in the space of complex χ-fields to go exactly through the saddle-point. The next task is to calculate corrections that originate from the vicinity of the saddle-point. At the saddle-point we choose the local orientation of the integration contour in the steepest descent way. Note that the steepest descent at a saddle-point may go along an imaginary or real direction. Finally, the integral in equation (46) is approximated by a Gaussian integral. The result of the Gaussian integration in equation (46), which leads to a correction in the saddle-point term, becomes

$$\sim \exp\left[-\mathcal{S}_0 - \mathcal{S}_1\right], \quad \mathcal{S}_1 = \frac{1}{2}\ln\left|\det\left[\hat{\Lambda}\right]\right|, \qquad \Lambda_{i\alpha;j\beta}(\chi^{(\mathrm{sp})}) \equiv \left.\frac{\partial^2 \mathcal{S}_0(\chi)}{\partial\chi_{i\alpha}\partial\chi_{j\beta}}\right|_{\chi=\chi^{(\mathrm{sp})}}, \tag{63}$$

where all the expressions are taken at $\chi^{(\mathrm{sp})}$.

Calculating the matrix of the second-order derivatives directly from equation (45), one arrives at

$$\forall \quad i \in \alpha: \quad \Lambda_{i\alpha;i\alpha}(\chi^{(\mathrm{sp})}) = \frac{2-q_i}{q_i-1}\left[1-m_i^2\right], \tag{64}$$

$$\forall \quad \alpha,\beta \ni i, \quad \alpha \neq \beta: \quad \Lambda_{i\alpha;i\beta}(\chi^{(\mathrm{sp})}) = \frac{1-m_i^2}{q_i-1}, \tag{65}$$

$$\forall \quad i,j \in \alpha, \quad i \neq j: \quad \Lambda_{i\alpha;j\alpha}(\chi^{(\mathrm{sp})}) = -\mu_{ij}, \tag{66}$$

where all matrix elements between the pairs/links $\{i,\alpha\}$ and $\{j,\beta\}$ sharing neither a common bit nor a common factor/check node are zero. In equations (64)–(66), m_i and μ_{ij} are the magnetizations and irreducible correlation functions, respectively, that are calculated within the saddle-point (BP) approximation (see equations (61) and (62)). Direct calculation of the Gaussian integrals based on the problem-specific information on the BP saddle-point solutions and the quadratic form matrix (64), (66), the latter also depending on the bare saddle-point solutions, provides a straightforward algebraic way of computing the partition function, magnetization, and any other spin-related objects within the Gaussian approximation.

A potential difficulty in the evaluation of Gaussian integrals originates from a relatively complex structure of the matrix $\hat{\Lambda}$. Specifically, the off-diagonal (66) term proportional to the irreducible pair correlation function induces coupling between different blocks related to the corresponding bits. In section 3.4.1 we analyse the Gaussian integrals perturbatively by expanding the off-diagonal term in an infinite series.

We will show that the only surviving terms in this expansion, after the Gaussian integrations are performed, correspond to loops in the factor graphs. We will demonstrate that the actual expansion parameter is the product over the loop of the terms $\mu_{ij}/(1-m_i^2)$, evaluated at the saddle-point.

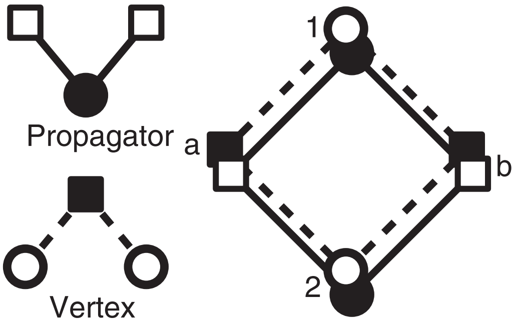

Figure 2. Gaussian approximation about the BP saddle-point. Diagrams for the propagator and the vertex are shown on the left. The right plot illustrates a loop contribution. The example corresponds to a loop in the model shown in the upper left corner of figure 1. A leg of a vertex (dashed line) should pair with a leg of a propagator (solid line). No unpaired legs are allowed.

3.4.1. Loop expansion in the Gaussian case. We start by introducing a convenient notation for the Gaussian integration:

$$\langle A\rangle_{\mathrm{bd}} \equiv \frac{\int\left(\prod_{p,\gamma}^{\gamma\in p}\mathrm{d}\zeta_{p\gamma}\right) A\exp[-(1/2)\zeta_{n\beta}\Lambda^{(\mathrm{bd})}_{n\beta;p\gamma}\zeta_{p\gamma}]}{\int\left(\prod_{p,\gamma}^{\gamma\in p}\mathrm{d}\zeta_{p\gamma}\right)\exp[-(1/2)\zeta_{n\beta}\Lambda^{(\mathrm{bd})}_{n\beta;p\gamma}\zeta_{p\gamma}]}, \tag{67}$$

where $\hat{\Lambda}^{(\mathrm{bd})}$ is the block (bit)-diagonal part $\hat{\Lambda}^{(\mathrm{off})}$ of $\hat{\Lambda}$ defined by equations (64) and (66) with the off-block-diagonal part of $\hat{\Lambda}$, described by equation (65) being ignored.

An important object $P_{i\alpha;j\beta} = \langle\zeta_{i\alpha}\zeta_{j\beta}\rangle_{\mathrm{bd}}$ in the expansion with respect to $\hat{\Lambda}^{(\mathrm{off})}$ will be referred to as a propagator following the traditional physics jargon of the Feynmann diagram expansion. It follows from equations (64) and (65) that the only nonzero component of the propagator is represented by

$$\alpha\neq\beta, \qquad \alpha,\beta\ni i: P_{i\alpha;i\beta} = \frac{1}{1-m_i^2}. \tag{68}$$

Note that the propagator has an interesting 'fermionic-repulsive' feature: for a fixed bit i it is strictly zero for coinciding factor/check indices, i.e., $P_{i\alpha;i\alpha}=0$. In addition to the 'propagator', the off-block-diagonal term is represented by a 'vertex'. The vertex term is nonzero only for

$$i\neq j, \qquad i,j\in\alpha: V_{i\alpha;i\beta} = \mu_{ij}. \tag{69}$$

It is also convenient to introduce a graphical notation for both the propagator and the vertex (see the left part of figure 2).

The correction to the partition function adopts the following form:

$$Z_g = \left\langle \exp\left[-\frac{1}{2}\zeta_{n\beta}\Lambda^{(\mathrm{off})}_{n\beta;p\gamma}\zeta_{p\gamma}\right]\right\rangle_{\mathrm{bd}} = \sum_{n=0}^{\infty}\frac{1}{2^n n!}\left\langle\left[-\zeta_{l\beta}\Lambda^{(\mathrm{off})}_{l\beta;p\gamma}\zeta_{p\gamma}\right]^n\right\rangle_{\mathrm{bd}}, \tag{70}$$

where the full partition function of the model is approximated as $Z\approx Z_0Z_g$. Each term in the sum on the rhs of equation (70) can be represented by a diagram. For an nth order term the diagram contains n-vertices. The Gaussian integration that corresponds to each term is performed in the following way. We first consider all possible Wick decompositions of the product of $2n$ ζ-terms in n pairs. Each pair in the product results in the corresponding propagator. The nth-order term on the rhs of equation (70) naturally decomposes into a sum of $n(n-1)$ terms each equal to a product of n-propagators and n-vertices. The key observation is that only very few of the terms survive due to the specific structure of the propagators (73) and the vertices (74). Indeed, only those terms do not vanish that consist of the propagators and vertices, coupled through their legs and forming a loop in the model factor graph. The structure is illustrated in figure 2. It is common for the Feynman diagrammatic techniques that the natural object is the logarithm $\ln(Z_g)$ of the partition function, since only connected diagrams, i.e., the ones that cannot be decomposed into a product of other diagrams, contribute to the object. This results in

$$\ln(Z_g) = \sum_C \frac{\prod_{\alpha\in C}^{i,j\in\alpha} \mu_{ij}}{\prod_{i\in C}(1-m_i^2)}, \tag{71}$$

where loops C are defined as closed directed self-avoiding paths in the model factor graph that pass from a bit to a factor/check and then from a factor/check to a bit, etc, in such a way that returns from a check that belongs to the path back to the preceding bit are not allowed. An example of a loop is shown in figure 2.

3.5. Loop calculus via integral representation

The Gaussian fluctuations analysis is justified only if the higher-order (third-, fourth-, etc) corrections to the Gaussian approximation are small compared to the major saddle-point and Gaussian contributions, and the expansion is controlled by some parameter. Jumping ahead we know that the loop expansion exposed by the Gaussian approximation is the correct one, in the sense that the connected loops contributions (no branching) provides the leading correction with respect to the branching parameter. However, to see how this general loop expansion actually works we need to expand the effective action to all orders around the saddle-point and classify an infinite number of perturbative terms, which seems a nightmare.

Fortunately, there is a way out of this technical problem that allows us to account for all-order corrections simultaneously. The method is based on introducing a set of new variables $\zeta_{i\alpha} \equiv \chi_{i\alpha} - \chi_{i\alpha}^{(\mathrm{sp})}$ followed by explicit decomposition of the integrand in equation (45) as a product of two non-Gaussian (with respect to the fields $\zeta_{i\alpha}$) terms, which are diagonal in the bit and factor/check representations, respectively:

$$Z \sim \int \left[\prod_i \prod_\alpha \mathrm{d}\zeta_{i\alpha}\right] \prod_i P_i\left(\zeta;\chi^{(\mathrm{sp})}\right) \prod_\alpha V_\alpha\left(\zeta;\chi^{(\mathrm{sp})}\right), \tag{72}$$

$$P_i \equiv \frac{\prod_{\alpha\ni i}\left[\sum_{\pi_\alpha} f_\alpha(\pi_\alpha)\exp\left(\sum_{j\in\alpha}\pi_\alpha^{(j)}\chi_{j\alpha}^{(\mathrm{sp})}\right)\exp\left(\pi_\alpha^{(i)}\zeta_{i\alpha}\right)\right]}{\left[\cosh\left(\sum_{\alpha\ni i}(\chi_{i\alpha}^{(\mathrm{sp})}+\zeta_{i\alpha})/(q_i-1)\right)\right]^{q_i-1}}, \tag{73}$$

$$V_\alpha \equiv \frac{\sum_{\pi_\alpha} f_\alpha(\pi_\alpha)\exp\left(\sum_{j\in\alpha}\pi_\alpha^{(j)}\left[\chi_{i\alpha}^{(\mathrm{sp})}+\zeta_{i\alpha}\right]\right)}{\prod_{i\in\alpha}\left[\sum_{\pi_\alpha} f_\alpha(\pi_\alpha)\exp\left(\sum_{j\in\alpha}\pi_\alpha^{(j)}\chi_{j\alpha}^{(\mathrm{sp})}\right)\exp\left(\pi_\alpha^{(i)}\zeta_{i\alpha}\right)\right]}. \tag{74}$$

Here we introduced the same factor to the numerator of P and denominator of V, respectively, that is local both in bit and check representations. The rationale behind such a decomposition choice is to ensure that in the case of the Gaussian approximate perturbative analysis of the effective action all correlations within a block associated with the same bit are included in the 'propagator' term P, while the inter-bit interaction appear only in the 'vertex' counterpart V.

We further introduce a set of convenient notations that will allow us to substantially simplify the calculations:

$$\forall\ \alpha \neq \beta : \langle A(\pi_\alpha)B(\pi_\beta)\rangle_\pi = \langle A(\pi_\alpha)\rangle_{\pi_\alpha}\langle B(\pi_\beta)\rangle_{\pi_\beta}, \tag{75}$$

$$\Big\langle A(\pi_\alpha)\Big\rangle_{\pi_\alpha} \equiv \frac{\sum_{\pi_\alpha} A(\pi_\alpha) f_\alpha(\pi_\alpha)\exp\left(\sum_{i\in\alpha}\pi_\alpha^{(i)}\chi_{i\alpha}^{(\mathrm{bp})}\right)}{\sum_{\pi_\alpha} f_\alpha(\pi_\alpha)\exp\left(\sum_{i\in\alpha}\pi_\alpha^{(i)}\chi_{i\alpha}^{(\mathrm{sp})}\right)}. \tag{76}$$

Using the notation, the 'propagator' and 'vertex' terms can be recast as follows:

$$P_i = \frac{\prod_{\alpha\ni i}\left[\cosh\zeta_{i\alpha} + m_i\sinh\zeta_{i\alpha}\right]}{\left[\cosh\left(\sum_{\alpha\ni i}(\chi_{i\alpha}^{(\mathrm{sp})}+\zeta_{i\alpha})/(q_i-1)\right)\right]^{q_i-1}}, \tag{77}$$

$$V_\alpha = \left[\sum_{\pi_\alpha} f_\alpha(\pi_\alpha)\exp\left(\sum_{i\in\alpha}\pi_\alpha^{(i)}\chi_{i\alpha}^{(\mathrm{sp})}\right)\right]\left\langle\prod_{i\in\alpha}\left(1+\frac{(\pi_\alpha^{(i)}-m_i)\tanh\zeta_{i\alpha}}{1+m_i\tanh\zeta_{i\alpha}}\right)\right\rangle_{\pi_\alpha}. \tag{78}$$

Of course, the number of terms in the series will grow exponentially with the size, very much like in the original formulation of the problem. However, wise classification of the terms followed by selecting (and calculating) a small number of relevant terms allows us not only to extract the BP approximation, but more importantly the leading-order corrections to BP. The leading-order corrections/terms will be associated with shortest loops on the Tanner graph, and this transparent geometrical interpretation will be coming through diagrammatic representation of the perturbative terms. Note that the diagrammatic technique we develop here is of a special kind. The major peculiarity of the technique is the non-Gaussian form of the P-term in equation (74). Our approach is technically reminiscent of the celebrated Vaks–Larkin–Pikin approach [23], used to calculate non-perturbative corrections to the ferromagnetic ground state in magnets. The reference is not precise as it only means to emphasize a vague structural relation of our method to the one introduced in the classical paper [23] where the 'propagator' P-term was also non-Gaussian.

We are now in a position to discuss a typical structure of the integrals over ζ for individual terms (diagrams). We notice that an individual integral over all possible ζ variables always decomposes into a product of independent integrals, each over the block of variables related to a bit. The simplest integral corresponds to a bit with all edges connected to it being uncoloured:

$$\begin{aligned} I_{0;i} &\equiv \int\prod_{\alpha\ni i}\mathrm{d}\chi_{i\alpha}P_i\left(\frac{1+m_i}{2}\right)^{q_i}\exp\left[-\sum_{\alpha\ni i}\chi_{i\alpha}^{(\mathrm{sp})}\right]+\left(\frac{1-m_i}{2}\right)^{q_i}\exp\left[\sum_{\alpha\ni i}\chi_{i\alpha}^{(\mathrm{sp})}\right] \\ &= \frac{(1-m_i^2)^{(q_i-1)/2}}{2^{q_i-1}} = \frac{\cosh\left(\sum_{\alpha\ni i}\chi_{i\alpha}^{(\mathrm{sp})}/(q_i-1)\right)^{1-q_i}}{2^{q_i-1}}, \end{aligned} \tag{79}$$

where one uses the saddle-point relation

$$\exp\left(\sum_{\alpha\ni i}\chi_{i\alpha}^{(\mathrm{sp})}\right)=\left(\frac{1+m_i}{1-m_i}\right)^{(q_i-1)/2}. \tag{80}$$

Combining equations (79), (80) and substituting the result in equation (72), we obtain for the first term in the series for the partition function

$$Z_0\sim\frac{\prod_\alpha\sum_{\pi_\alpha}f_\alpha(\pi_\alpha)\exp\left(\sum_{i\in\alpha}\pi_\alpha^{(i)}\chi_{i\alpha}^{(\mathrm{sp})}\right)}{\prod_i\cosh\left(\sum_{\alpha\ni i}\chi_{i\alpha}^{(\mathrm{sp})}/(q_i-1)\right)^{1-q_i}}, \tag{81}$$

which exactly reproduces the aforementioned saddle-point result.

For the general p-order term one arrives at

$$\begin{aligned} I_{p;i,\{\alpha_l;\ l=1,\ldots,p,\ \alpha_l\in i\}}&\equiv\int\prod_{\alpha\ni i}\mathrm{d}\chi_{i\alpha}\,P_i\prod_{l=1}^{p}\frac{\tanh\zeta_{i\alpha_l}}{(1+m_i\tanh\zeta_{i\alpha_l})}\\ &=\frac{\exp\left[-\sum_{\alpha\ni i}\chi_{i\alpha}^{(\mathrm{sp})}\right]}{2^{q_i}}(1+m_i)^{q_i-p}+(-1)^p\frac{\exp\left[\sum_{\alpha\ni i}\chi_{i\alpha}^{(\mathrm{sp})}\right]}{2^{q_i}}(1-m_i)^{q_i-p}\\ &=I_{0;i}\frac{(1-m_i)^{p-1}+(-1)^p(1+m_i)^{p-1}}{2(1-m_i^2)^{p-1}}. \end{aligned} \tag{82}$$

The resulting expression for the entire series derived directly from equations (72), (77), (78) and equation (82) is given by equations (8)–(10).

4. Conclusions

We conclude by presenting a brief outline for our ongoing and future research activities on the way of extending the loop calculus detailed in this paper.

Gauge invariance of the vertex models has been discussed above in the context of two specific representations we utilized to derive the loop series formula. However, this important notion allows more universal and mathematically accurate formulation. For this purpose, it is convenient to introduce the notion of a graphic tensor and corresponding graphic trace (convolution). The graphic trace concept generalizes the standard (in statistical mechanics) transfer matrix approach to the models defined using arbitrary graphs. This allows us to formulate the loop expansion, and the BP equations, as well as the Bethe free energy, in a gauge-invariant form. The loop expansion becomes nothing more than a representation of the partition function as a sum over all possible configurations using a special BP gauge. The graphical trace and the unifying gauge-invariant approach will be discussed in detail elsewhere [24]. The universal formulation allows for a natural and straightforward extension of the loop calculus to more general statistical models that operate with non-binary alphabets (e.g. Potts model on a graph). This general problem will also be discussed in [24].

More generally, we anticipate that the loop calculus can be extended to any classical models residing in graphs that are formulated in terms of continuous fields of Abelian and non-Abelian origin, e.g. O(2) and O(3) models, respectively. Moreover, the approach should also work for quantum models and fields, e.g. the quantum Heisenberg model on a graph. The latter may be of substantial interest for developing new approaches in quantum information theory.

A loop series offers an exact representation for the partition function, and also correlation functions, that can be used for improving approximate algorithms. This should be understood as follows. Many problems in statistical physics, information and computer sciences are intractable, in the sense that the number of steps required to accomplish a computation (i.e., to calculate an observable) grows exponentially with the system size. Then the issue of an approximation and related approximate computational algorithm emerges. Development of a sequence of approximations with gradually increasing complexity becomes an important task. On the one hand, the higher is the term in the sequence, the better it approximates the full answer. On the other hand, the complexity should be linear or polynomial for at least some number of low-order terms. Given that the first term in the loop series is the BP term, which is known to constitute already a very efficient approximation/algorithm, one can use the higher-order loop corrections as a regular way of BP algorithm improvement.

The loop series also introduces an explicit BP measure on the graph: any loop contribution can be expressed in terms of local objects, magnetizations and irreducible correlation functions calculated within BP. Therefore, looking for individual loop contributions that dominate the correction to the bare BP approximation constitutes a particularly attractive and computationally feasible strategy [25]. As a side remark we note that various options are available for calculating the bare BP contribution and estimating the magnetizations and irreducible correlation functions within BP. First of all, one may use the original iterative algorithm of Gallager [7]. The linear programming approach of [26] is another very attractive possibility at large signal-to-noise ratios in the cases when the iterative BP does not converge. Finally, one may develop an iterative relaxation algorithm that is guaranteed to converge to a true minimum of the Bethe free energy [27]. Such an algorithm is expected to perform better than the linear programming algorithm at finite temperatures.

We anticipate the proposed scheme to work very well in many cases, especially for graphs that are locally tree-like. The models with long loops emerge naturally in the context of decoding of LDPC codes [7, 10, 11] and also in the K-SAT satisfiability problem in computer science [13]–[15].

The loop series can also be very useful for theoretical analysis of problems with disorder [28], e.g. of the random graph type [29, 30]. The goal here is to calculate the loop corrections to various disorder-averaged correlation functions. A particularly interesting question is how to differentiate contributions that originate from loops of different sizes. Depending on the regime one may expect either dominance of some limited number of shortest loops, or a distributed effect of many loops. Thus, for the Viana–Bray model [29], which contains a large number of short loops, considered in the high-temperature regime, the latter possibility was reported in the formal $1/N$ replica expansion [16]. The leading correction to the BP expression for the averaged free energy is dominated by a combined effect of many long loops. Further analysis of this and other models, especially the ones corresponding to expurgated ensembles of random graphs modelling LDPC codes with large girth [31], is required to clarify the statistical role of loops of different lengths.

Finally, we are optimistic about using the loop calculus developed in this paper for further analysis and algorithmic exploration of the standard lattice models, i.e., regular structures with many short loops. A particularly interesting yet challenging direction of research would be using the loop calculus, which naturally differentiates the loops of different sizes and shapes, for analysis of the critical point behaviour in the lattice models.

Acknowledgments

We are thankful to M Stepanov for many fruitful discussions. The work at LANL was supported by the LDRD program, and through start-up funds at WSU.

Appendix A: Bethe free energy

In this appendix we reproduce a derivation of the Belief–Propagation equation based on the Bethe free energy variational principle, following closely the description of [6], e.g. translating it to our notations. We describe the Bethe free energy approach for the factor graph model and general vertex models in the two subsequent subsections, respectively.

Appendix A.1. Bethe free energy for the factor graph model

In this approach trial probability distributions, called beliefs, are introduced both for bits and checks b_i and b_α, respectively, where $i = 1, \ldots, N$, $\alpha = 1, \ldots, M$. Each belief depends on the corresponding spin realization. Thus, a belief at a bit actually consists of two probabilities, $b_i(+)$ and $b_i(-)$, and we use a natural notation $b_i(\sigma_i)$. There are 2^k beliefs defined at a check, k being the number of bits connected to the check, and we introduce vector notation $\boldsymbol{\sigma}_\alpha = (\sigma_{i_1}, \ldots, \sigma_{i_k})$ where $i_1, \ldots, i_k \in \alpha$ and $\sigma_i = \pm 1$. Beliefs, as corresponding probabilities, satisfy the following inequality constraints:

$$0 \le b_i(\sigma_i), \qquad b_\alpha(\boldsymbol{\sigma}_\alpha) \le 1, \tag{A.1}$$

the normalization constraints

$$\sum_{\sigma_i} b_i(\sigma_i) = \sum_{\boldsymbol{\sigma}_\alpha} b_\alpha(\boldsymbol{\sigma}_\alpha) = 1, \tag{A.2}$$

as well as the consistency (between bits and checks) constraints

$$\sum_{\boldsymbol{\sigma}_\alpha \backslash \sigma_i} b_\alpha(\boldsymbol{\sigma}_\alpha) = b_i(\sigma_i), \tag{A.3}$$

where $\boldsymbol{\sigma}_\alpha \backslash \sigma_i$ stands for all possible configurations of σ_j with $j \in \alpha$, $j \neq i$.

The Bethe free energy is defined as a difference of the Bethe self-energy and the Bethe entropy,

$$F_{\text{Bethe}} = U_{\text{Bethe}} - H_{\text{Bethe}}, \tag{A.4}$$

defined as

$$U_{\text{Bethe}} = -\sum_\alpha \sum_{\boldsymbol{\sigma}_\alpha} b_\alpha(\boldsymbol{\sigma}_\alpha) \ln f_\alpha(\boldsymbol{\sigma}_\alpha), \tag{A.5}$$

$$H_{\text{Bethe}} = -\sum_\alpha \sum_{\boldsymbol{\sigma}_\alpha} b_\alpha(\boldsymbol{\sigma}_\alpha) \ln b_\alpha(\boldsymbol{\sigma}_\alpha) + \sum_i (q_i - 1) \sum_{\sigma_i} b_i(\sigma_i) \ln b_i(\sigma_i), \tag{A.6}$$

where $\boldsymbol{\sigma}_\alpha = (\sigma_{i_1}, \ldots, \sigma_{i_k})$, $i_1, \ldots, i_k \in \alpha$ and $\sigma_i = \pm 1$. The entropy term for a bit enters equation (A.4) with the coefficient $1 - q_i$ to account for the right counting of the number of configurations for a bit: if all entries for a bit (e.g. into the check term) are counted the total counting should give $+1$ for the bit.

Note that the definition of f_α according to equation (A.5) is not unique. A convenient choice of the factor function describing an LDPC code would be

$$f_\alpha(\boldsymbol{\sigma}_\alpha) \equiv \exp\left(\sum_{i\in\alpha} h_i\sigma_i/q_i\right)\delta\left(\prod_{i\in\alpha}\sigma_i, 1\right). \tag{A.7}$$

Optimal configurations of beliefs are the ones that minimize the Bethe free energy (A.4) subject to the constraints (A.1)–(A.3). Introducing the constraints as the Lagrange multiplier term to the effective Lagrangian

$$\begin{aligned} L = F_{\text{Bethe}} &+ \sum_\alpha \gamma_\alpha\left(\sum_{\boldsymbol{\sigma}_\alpha} b_\alpha(\boldsymbol{\sigma}_\alpha) - 1\right) + \sum_i \gamma_i\left(\sum_{\sigma_i} b_i(\sigma_i) - 1\right) \\ &+ \sum_i\sum_{\alpha\ni i}\sum_{\sigma_i}\lambda_{i\alpha}(\sigma_i)\left(b_i(\sigma_i) - \sum_{\boldsymbol{\sigma}_\alpha\setminus\sigma_i} b_\alpha(\boldsymbol{\sigma}_\alpha)\right), \end{aligned} \tag{A.8}$$

and looking for the extremum with respect to all possible beliefs leads to

$$\frac{\delta L}{\delta b_a(\boldsymbol{\sigma}_a)} = 0 \Rightarrow \quad b_\alpha(\boldsymbol{\sigma}_\alpha) = f_\alpha(\boldsymbol{\sigma}_\alpha)\exp\left[-\gamma_\alpha - 1 + \sum_{i\in\alpha}\lambda_{i\alpha}(\sigma_i)\right], \tag{A.9}$$

$$\frac{\delta L}{\delta b_i(\boldsymbol{\sigma}_i)} = 0 \Rightarrow \quad b_i(\sigma_i) = \exp\left[\frac{1}{q_i - 1}\left(\gamma_i + \sum_{\alpha\ni i}\lambda_{i\alpha}(\sigma_i)\right) - 1\right]. \tag{A.10}$$

Substituting $\lambda_{i\alpha}(\sigma_i) \equiv \ln\prod_{\beta\ni i;\beta\neq\alpha}\mu_{i\beta}(\sigma_i)$ into equations (A.9) and (A.10) we arrive at

$$b_\alpha(\boldsymbol{\sigma}_\alpha) \propto f_\alpha(\boldsymbol{\sigma}_\alpha)\prod_{i\in\alpha}\prod_{\beta\ni i}^{\beta\neq\alpha}\mu_{i\beta}(\sigma_i), \tag{A.11}$$

$$b_i(\sigma_i) \propto \prod_{\alpha\ni i}\mu_{i\alpha}(\sigma_i), \tag{A.12}$$

where $\propto$ is used to indicate that we should use the normalization conditions (A.2) to guarantee that the beliefs sum up to one. Applying the consistency constraint (A.3) to equation (A.11), making the summation over all spins but the given σ_i, and also making use of equation (A.12) we derive the Belief–Propagation equations:

$$\prod_{\alpha\ni i}\mu_{i\alpha}(\sigma_i) \propto b_i(\sigma_i) \propto \left[\prod_{\beta\ni i}^{\beta\neq\alpha}\mu_{i\beta}(\sigma_i)\right]\sum_{\sigma_\alpha\setminus\sigma_i} f_\alpha(\sigma_\alpha)\prod_{j\in\alpha}^{j\neq i}\prod_{\beta\ni j}^{\beta\neq\alpha}\mu_{j\beta}(\sigma_j). \tag{A.13}$$

The rhs of equation (A.13) rewritten for the LDPC case (A.7) becomes

$$b_i(\sigma_i) \propto \exp[h_i\sigma_i]\left[\prod_{\beta\ni i}^{\beta\neq\alpha}\mu_{i\beta}(\sigma_i)\right]\left[\prod_{j\in\alpha}^{j\neq i}(\mu_{j\alpha}(+) + \mu_{j\alpha}(-)) + \sigma_i\prod_{j\in\alpha}^{j\neq i}(\mu_{j\alpha}(+) - \mu_{j\alpha}(-))\right]. \tag{A.14}$$

Thus constructing $b_i(+)/b_i(-)$ for the LDPC case in two different ways, corresponding to the left and right relations in equation (A.13), equating the results and introducing the $\eta_{i\alpha}$ field

$$\exp[2\eta_{i\alpha}] = \frac{\mu_{i\alpha}(+)}{\mu_{i\alpha}(-)}, \tag{A.15}$$

one arrives at the BP equations for the $\eta_{i\alpha}$ fields of the LDPC code:

$$\eta_{i\alpha} = h_i + \sum_{\beta\ni i}^{\beta\neq\alpha} \tanh^{-1}\left[\prod_{j\in\beta}^{j\neq i} \tanh\eta_{j\beta}\right]. \tag{A.16}$$

Appendix A.2. Bethe free energy for the general vertex model

The variational approximation for the model that generalizes the factor graph case discussed in appendix A.1, reads as follows. One minimizes the following Bethe free energy

$$F_{\mathrm{gvm}} = \sum_a \sum_{\boldsymbol{\sigma}_a} b_a(\boldsymbol{\sigma}_a) \ln\left(\frac{b_a(\boldsymbol{\sigma}_a)}{f_a(\boldsymbol{\sigma}_a)}\right) - \sum_{a,c}^{c\in a} \sum_{\sigma_{ac}} b_{ac}(\sigma_{ac}) \ln b_{ac}(\sigma_{ac}), \tag{A.17}$$

with respect to $b_a(\boldsymbol{\sigma}_a), b_{ac}(\sigma_{ac})$ fields under the conditions

$$\forall\ a, c;\ \ c \in a : 0 \leq b_a(\boldsymbol{\sigma}_a), b_{ac}(\sigma_{a,c}) \leq 1, \tag{A.18}$$

$$\forall\ a, c;\ \ c \in a : \sum_{\boldsymbol{\sigma}_a} b_a(\boldsymbol{\sigma}_a) = \sum_{\sigma_{a,c}} b_{ac}(\sigma_{a,c}) = 1, \tag{A.19}$$

$$\forall\ a, c;\ \ c \in a : b_{ac}(\sigma_{ac}) = \sum_{\boldsymbol{\sigma}_a\backslash\sigma_{ac}} b_a(\boldsymbol{\sigma}_a) = \sum_{\sigma_c\backslash\sigma_{ac}} b_c(\boldsymbol{\sigma}_c), \tag{A.20}$$

where as usual we assume $\sigma_{ac} = \sigma_{ca}$. The second term on the rhs of equation (A.17) is the entropy term which takes care of the 'double' counting' of the link contribution: any link enters twice in the entropy part of the first term on the rhs of equation (A.17).

Extension of these formulae to the orientable vertex model case is straightforward. It is achieved by partitioning the entire family of vertices $\{a\}$ into two subfamilies $\{i\}$ and $\{\alpha\}$. After that one just needs to replicate equations (A.17)–(A.19) in the bit and check versions respectively, while equation (A.21) adopts the following form:

$$\forall\ i, \alpha;\ \ i \in \alpha : \sum_{\boldsymbol{\sigma}_i\backslash\sigma_{i\alpha}} b_i(\boldsymbol{\sigma}_i) = \sum_{\boldsymbol{\sigma}_\alpha\backslash\sigma_{i\alpha}} b_\alpha(\boldsymbol{\sigma}_\alpha). \tag{A.21}$$

Furthermore, considering the case of the orientable vertex model and substituting a particular form of the $f_i(\boldsymbol{\sigma}_i)$ correspondent to equation (4), we find that equation (A.17) turns into equations (A.4)–(A.6) under a natural substitution:

$$b_i(\boldsymbol{\sigma}_i) = \begin{cases} b_i(\sigma_i), & \sigma_{i\alpha} = \sigma_{i\beta} \quad \forall\alpha, \beta \ni i \\ 0, & \text{otherwise.} \end{cases} \tag{A.22}$$

Appendix B. Single loop example

This appendix serves an illustrative purpose. We show directly how the loop formula (8) works for a simple example of the factor graph model (1) with a single loop (two bits and two checks, see figure B.1). For this simple model the Belief–Propagation equations (A.11) and (A.12) adopt the following form:

$$\text{for } \alpha,\beta = a,b; \quad \beta \neq \alpha : b_\alpha(\sigma_1,\sigma_2) = \frac{f_\alpha(\sigma_1,\sigma_2) d_{1\beta}^{\sigma_1/2} d_{2\beta}^{\sigma_2/2}}{\sum_{\sigma'_{1,2}} f_\alpha(\sigma'_1,\sigma'_2) d_{1\beta}^{\sigma'_1/2} d_{2\beta}^{\sigma'_2/2}}, \tag{B.1}$$

$$\text{for } i = 1,2 : b_i(\sigma_i) = \frac{d_{ia}^{\sigma_i/2} d_{ib}^{\sigma_i/2}}{\sum_{\sigma'_i} d_{ia}^{\sigma'_i/2} d_{ib}^{\sigma'_i/2}} \tag{B.2}$$

where the factor functions $f_a(\sigma_1,\sigma_2)$, $f_b(\sigma_1,\sigma_2)$, defined for $\sigma_{1,2} = \pm 1$ are considered to be arbitrary. Equation (B.2) is reduced to the set of quadratic equations that can be solved explicitly, yielding

$$\begin{aligned} d_{1a} = (&-f_a(-,-)f_b(-,-) - f_a(-,+)f_b(-,+) + f_a(+,-)f_b(+,-) + f_a(+,+)f_b(+,+) \\ &\times [4(f_b(-,-)f_a(-,+) + f_b(+,-)f_a(+,+))(f_a(-,-)f_b(-,+) \\ &+ f_a(+,-)f_b(+,+))(f_a(-,-)f_b(-,-) - f_a(-,+)f_b(-,+) \\ &+ f_a(+,-)f_b(+,-) - f_a(+,+)f_b(+,+))^2]^{1/2}) \\ &\times [2(f_a(-,-)f_b(+,-) + f_a(-,+)f_b(+,+))]^{-1}, \end{aligned} \tag{B.3}$$

where the BP expressions for d_{1b}, d_{2a} and d_{2b} can be derived by making proper permutations of indices and arguments in equation (B.3). Using these solutions we arrive at the following expressions for the partition function calculated within the BP approach:

$$\begin{aligned} Z_0 = &\frac{\prod_\alpha \sum_{\sigma_{1,2}}^{\beta\neq\alpha} f_\alpha(\sigma_1,\sigma_2) d_{1\beta}^{\sigma_1/2} d_{2\beta}^{\sigma_2/2}}{\prod_i \sum_{\sigma_i} d_{ia}^{\sigma_i/2} d_{ib}^{\sigma_i/2}} \\ &= \tfrac{1}{2}(f_a(-,-)f_b(-,-) + f_a(-,+)f_b(-,+) + f_a(+,-)f_b(+,-) \\ &\quad + f_a(+,+)f_b(+,+) + [4(f_b(-,-)f_a(-,+) + f_b(+,-)f_a(+,+)) \\ &\quad \times (f_a(-,-)f_b(-,+) + f_a(+,-)f_b(+,+)) + (f_a(-,-)f_b(-,-) \\ &\quad - f_a(-,+)f_b(-,+) + f_a(+,-)f_b(+,-) - f_a(+,+)f_b(+,+))^2]^{1/2}). \end{aligned} \tag{B.4}$$

Bit magnetizations as well as irreducible correlation functions at the checks are found upon direct substitution of (B.3) and similar expressions for the other d-variables in terms of the factor functions into

$$i = 1,2 : m_i = \sum_{\sigma_i} \sigma_i b_i(\sigma_i), \tag{B.5}$$

$$\alpha = a,b : \mu_\alpha = \sum_{\sigma_1,\sigma_2} (\sigma_1 - m_1)(\sigma_2 - m_2) b_\alpha(\sigma_1,\sigma_2). \tag{B.6}$$

Substituting these results, together with equation (B.4), into the loop expression equation (8) for the model partition function we obtain

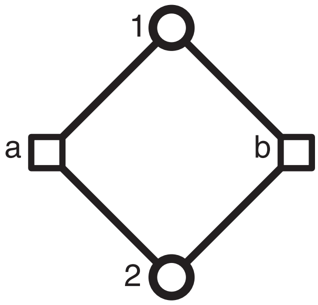

Figure B.1. Factor graph for the single loop model consisting of two nodes and two bits.

$$Z = Z_0\left(1 + \frac{\mu_a\mu_b}{(1-m_1^2)(1-m_2^2)}\right)$$
$$= f_a(-,-)f_b(-,-) + f_a(-,+)f_b(-,+) + f_a(+,-)f_b(+,-) + f_a(+,+)f_b(+,+), \tag{B.7}$$

which coincides with the exact expression for the model partition function that can be evaluated directly.

References

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

[2] Peierls R, 1936 *Proc. Camb. Phil. Soc.* **32** 477

[3] Baxter R J, 1982 *Exactly Solvable Models in Statistical Mechanics* (New York: Academic)

[4] Kikuchi R, *A theory of cooperative phenomena*, 1951 *Phys. Rev.* **81** 988

[5] Morita T, *Cluster variation method for non-uniform Ising and Heisenberg models and spin-pair correlation functions*, 1991 *Prog. Theor. Phys.* **85** 243

[6] Yedidia J S, Freeman W T and Weiss Y, *Constructing free energy approximations and generalized belief propagation algorithms*, 2005 *IEEE Trans. Inf. Theory* **51** 2282

[7] Gallager R G, 1963 *Low Density Parity Check Codes* (Cambridge, MA: MIT Press)

[8] Gallager R G, 1968 *Information Theory and Reliable Communication* (New York: Wiley)

[9] Sourlas N, 1989 *Nature* **339** 693

[10] MacKay D J C, *Good error-correcting codes based on very sparse matrices*, 1999 *IEEE Trans. Inf. Theory* **45** 399

[11] Richardson T and Urbanke R, *The renaissance of Gallager's low-density parity-check codes*, 2003 *IEEE Commun. Mag.* **41** 126

[12] Pearl J, 1988 *Probabilistic Reasoning in Intelligent Systems: Network of Plausible Inference* (San Francisco, CA: Kaufmann)

[13] Mezard M, Parisi G and Zecchina R, *Analytic and algorithmic solution of random satisfiability problems*, 2002 *Science* **297** 812

[14] Mezard M and Zecchina R, *Random K-satisfiability problem: from an analytical solution to efficient algorithm*, 2002 *Phys. Rev.* E **66** 056126

[15] Braunstein A and Zecchina R, *Survey propagation as local equilibrium equations*, 2004 *J. Stat. Mech.* P06007 [cond-mat/0312483]

[16] Montanari A and Rizzo T, *How to compute loop correction to Bethe approximation*, 2005 *J. Stat. Mech.* P10011 [cond-mat/0506769]

[17] Parisi G and Slanina F, *Loop expansion around the Bethe–Peierls approximation for lattice models*, 2006 *J. Stat. Mech.* P02003 [cond-mat/0512529]

[18] Efetov K B, *Effective medium approximation in localization theory: saddle-point in a Lagrangian formulation*, 1990 *Physica* A **167** 119

[19] Chertkov M and Chernyak V Y, 2006 *Phys. Rev.* E **73** 065102(R)

[20] Kschischang F R, Frey B J and Loeliger H-A, *Factor graphs and the sum–product algorithm*, 2001 *IEEE Trans. Inf. Theory* **47** 498

[21] Forney G D, *Codes on graphs: normal realizations*, 2001 *IEEE Trans. Inf. Theory* **47** 520

[22] Loeliger H-A, *An introduction to factor graphs*, 2001 *IEEE Signal Process. Mag.* (January) 28–41

[23] Vaks V G, Larkin A I and Pikin S A, 1966 *Zh. Eksp. Teor. Fiz.* **53** 1089
Vaks V G, Larkin A I and Pikin S A, *Spin waves and correlation functions in a ferromagnetic*, 1968 *Sov. Phys. JETP* **26** 647 (translation)

[24] Chernyak V Y and Chertkov M, *Gauge invariance and loop calculus for discrete statistical models* , 2006 in preparation

[25] Chertkov M, Chernyak V Y and Stepanov M G, *Improving belief propagation algorithm with loop calculus* in preparation

[26] Feldman J, Wainwright M and Karger D R, *Using linear programming to decode linear codes*, 2003 *Conf. on Information Sciences and Systems* (*The John Hopkins University, March 2003*)

[27] Stepanov M G and Chertkov M, *Dynamics of iterative decoding* , 2006 in preparation

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

[29] Viana L and Bray A, *Phase diagrams for dilute spin glasses*, 1985 *J. Phys. C: Solid State Phys.* **18** 3037

[30] Montanari A, *The glassy phase of Gallager codes*, 2001 *Eur. Phys. J.* B **23** 121

[31] Di C, Proietti D, Telatar I E, Richardson T J and Urbanke R L, *Finite-length analysis of low density parity check codes on the binary erasure channel*, 2002 *IEEE Trans. Inf. Theory* **48** 1570