Message-Passing Algorithms for Non-Linear Nodes and Data Compression

Stefano Ciliberti^a Marc Mézard^a Riccardo Zecchina^b

^aLaboratoire de Physique Théorique et Modèles Statistiques, Université de Paris-Sud, Orsay, France; ^bICTP, Trieste, Italy

Key Words

Source and channel coding • Networks • Random graphs • Survey propagation

Abstract

The use of parity-check gates in information theory has proved to be very efficient. In particular, error correcting codes based on parity checks over low-density graphs show excellent performances. Another basic issue of information theory, namely data compression, can be addressed in a similar way by a kind of dual approach. The theoretical performance of such a parity source coder can attain the optimal limit predicted by the general rate-distortion theory. However, in order to turn this approach into an efficient compression code (with fast encoding/decoding algorithms) one must depart from parity checks and use some general random gates. By taking advantage of analytical approaches from the statistical physics of disordered systems and SP-like message passing algorithms, we construct a compressor based on low-density non-linear gates with a very good theoretical and practical performance.

Simplexus

A new data compression protocol could emerge from an adaptation of statistical mechanics described by Ciliberti, Mezard, and Zecchina. The method exploits a new class of algorithms for solving 'constraint satisfaction problems' (CSP). In the future, it could have practical applications in information theory and image processing.

The researchers explain how parity-check gates in information theory are a very efficient way of ensuring the integrity of a data stream. Indeed, they are particularly effective and perform well in correcting code. However, information theory is not only about correction, there is the issue of compression to consider too. Ciliberti, Mezard, and Zecchina suggest that data compression might also be handled using a similar approach to data correction. Indeed, a parity source coder could, in theory, attain the optimal level of compression, whether it could be realized in practice is a different matter and one which the researchers hope to address in the present paper.

There is an increasingly apparent connection between statistical physics and modern error correcting codes, known as the low-density parity check (LDPC) codes, explain the researchers. In such codes, the choice of the encoding schemes maps into a graphical model and the LDPC codes exploit redundancy by adding bits which must satisfy some random sparse linear set of equations – the parity checks, in other words. Such equations are used in the decoding phase to reconstruct the original codeword (and hence the message) from a corrupted signal.

This connection suggested to the team that they might turn to the analytical methods of statistical physics for disordered systems, which has common roots with information theory, to develop an efficient compression code that is both fast at encoding and decoding data. The essential tool here is the generalization of a standard message-passing procedure widely used in

Copyright © 2006 S. Karger AG, Basel

Fax +41 61 306 12 34
E-Mail karger@karger.ch
www.karger.com

KARGER

© 2006 S. Karger AG, Basel
1424–8492/06/0033–0058
\$23.50/0
Accessible online at:
www.karger.com/cpu

Stefano Ciliberti, Laboratoire de Physique Théorique et Modèles Statistiques, bâtiment 100, Université de Paris-Sud
FR–91405 Orsay Cedex (France)
Tel. +33 1 69 15 73 24, Fax +33 1 69 15 65 25
E-Mail ciliberti@roma1.infn.it

1 Introduction

Information theory and statistical physics have common roots. In recent years, this interconnection has found further confirmation in the analysis of modern error correcting codes, known as low density parity check (LDPC) codes [1–3], by statistical physics methods [4–6]. In such codes, the choice of the encoding schemes maps into a graphical model: the way in which LDPC codes exploit redundancy is by adding bits which are bound to satisfy some random sparse linear sets of equations – the so-called parity checks. Such equations are used in the decoding phase to reconstruct the original codeword (and hence the message) from the corrupted signal. The parity checks can be represented by a graph indicating which variables participate in each check. The randomness and the sparsity of the parity checks are reflected into the characteristics of the associated graphs, a fact that makes mean-field-type statistical physics methods directly applicable. Thanks to the duality between channel coding and source coding [3], similar constructions can be used to perform both lossless and lossy data compression [7, 8]. It must be mentioned that while there exist some practical algorithms for the lossless source coding which are very efficient [9–11], much less work has been done so far for the lossy case [12]. In this work we present a new coding technique which consists of a generalization of parity-check codes for lossy data compression to non-linear codes [some of the results have appeared in 13]. This issue has been addressed recently using methods from the statistical physics of disordered systems: a non-linear perceptron [14] or a satisfiability problem [15] have been used to develop practical source coding schemes. Other recent advances in the field can be found in Garcia-Frias and Zhao [16], Caire et al. [17] and Matsunaga and Yamamoto [18].

We consider the following problem. We have a random string of uncorrelated bits $x_1, \dots, x_M$ ($\text{prob}(x_a = 0) = \text{prob}(x_a = 1) =$

$1/2$) and we want to compress it to the shorter coded sequence $\sigma_1, \dots, \sigma_N$ (encoding). The rate R of the process is $R = N/M$. Once we recover the message (decoding), we may have made some errors and thus we are left in principle with a different sequence $x^*_1, \dots, x^*_M$. The number of different bits normalized by the length of the string is the distortion,

$$D = \frac{1}{M} \sum_{a=1}^M (1 - \delta(x_a, x_a^*)) . \quad (1)$$

The Shannon theorem states that the minimum rate at which we can achieve a given distortion D is $R_{\min} = 1 - H_2(D)$, where $H_2(D)$ is the binary entropy of the probability distribution used to generate the x_a 's. This $R_{\min}$ is the value one has to compare to in order to measure the performance of a compression algorithm. This problem is a simple version of the lossy data compression problem. In real applications one is often more interested in compression through quantization of a signal with letters in a larger alphabet (or continuous signal [19]). Here we shall keep to the memoryless case of uncorrelated unbiased binary sources, and we hope to be able to generalize the present approach to more realistic cases in the near future.

1.A Constraint Satisfaction Problems

The underlining structure of the protocol that we are going to introduce is provided by a constraint satisfaction problem (CSP). A CSP with N variables is defined in general by a cost function

$$E[\sigma] = \sum_{a=1}^M \varepsilon_a(\{\sigma_i \in a\}) , \quad (2)$$

where the function node $\varepsilon_a(\sigma_1, \dots, \sigma_{K_a})$ involves a subset of K_a variables randomly chosen among the whole set. The problem is fully defined by choosing some ε_a and an instance is specified by the actual variables involved in each constraint. A useful

computer science, the belief propagation (BP). The idea behind the method is simple. One writes down local iterative equations for a single variable for a given status of its neighbours. Due to the diluted graph geometry, the total set of equations can be self-consistently closed and, most importantly, solved by numerical (fast) iteration. This method, successful in many problems, has to be generalized in order to deal with the CSP which are relevant for data compression. In the corresponding 'survey propagation' (SP) message passing procedure, one has to solve iterative equations for the probability distributions of the variables – a more complex object. In order to have a fast implementation of the encoding/decoding steps, however, the team was forced to depart from the initial idealistic approach based on parity checks and to use general random gates. As such, their compression system utilizes low-density non-linear gates and demonstrates very good theoretical and practical performance.

In correction, parity checks can be represented by a graph indicating which variables participate in each check and the randomness and sparsity of the parity checks is reflected in the graphs. The fact that there is a duality between channel coding and source coding means that similar constructions might be used to perform both lossless and lossy data compression.

In compression, every check is controlled by one bit of the string one wants to compress. Data compression essentially distils to a single problem: to reduce a random string of bits to a shorter coded string, one that would fill less space in computer memory.

The compression rate of the process is a measure of the size reduction and is just given by the ratio between the encoded and original string. Decoding the message has to allow retrieval of the original string, but errors can occur, thus leading to a distortion. A good compression algorithm should

representation for a CSP is the factor graph (fig. 1a). The two kinds of nodes in this graph are the variables and the constraints. For the sake of simplicity we shall take the functions ε_a to have values in $\{0, 2\}$ (the clause is, respectively, SAT or UNSAT) and the variables σ_i will be Ising spins ($\sigma_i = \pm 1$). A somewhat special case of CSP which has been studied extensively occurs when $K_a = K$ for any a , and we shall restrict ourselves to this case in what follows. One can then show that the degree of a given variable is Poisson distributed with mean value KM/N and that the typical length of a loop is $\mathcal{O}(\log N/\log(KM/N))$. We are interested in the limit $M, N \rightarrow \infty$, where $\alpha \equiv M/N$ is fixed and plays the role of a control parameter. Once a problem is given in terms of its graph representation, an instance corresponds to a given graph chosen among all the legal ones with uniform probability. We will go back to this problem in section 2. We first show how a CSP can be used as a tool for compressing data (fig. 1).

1.B Encoding

We use the initial word $x_1, x_2, \dots, x_M$ to construct a set of M constraints between N

Boolean variables $\sigma_1, \sigma_2, \dots, \sigma_N$. The factor graph is supposed to be given. Each constraint a involves K_a variables and is defined by two complementary subsets of configurations S_0^a and S_1^a . The value of x_a controls the role of the subsets as follows: If $x_a = 0$ then all the configurations $\{\sigma_1, \dots, \sigma_{K_a}\} \in S_0^a$ satisfy the constraint, i.e. $\varepsilon_a = 0$; all the configurations $\{\sigma_1, \dots, \sigma_{K_a}\} \in S_1^a$ do not satisfy the constraint and thus $\varepsilon_a = 2$. If $x_a = 1$ the role of the two subsets is exchanged. This defines the CSP completely and we can look for a configuration of σ_i that minimizes the number of violated constraints. This is the encoded word and the rate of the process is $R = 1/\alpha$.

1.C Decoding

Given the configuration $\sigma_1, \sigma_2, \dots, \sigma_N$, we compute for each node a whether the variables $\sigma_{i_1}^a, \dots, \sigma_{i_{K_a}}^a$ are in S_0 (in this case we set $x_a^* = 0$) or in S_1 (leading to $x_a^* = 1$). Since a cost $\varepsilon = 2$ is paid for each UNSAT clause, the energy is related to the distortion by

$$D = \frac{E}{2M} = \frac{E}{2\alpha N} = R \frac{E}{2N}. \quad (3)$$

be able to reduce the string length a lot while retaining all the information contained in it, thus giving a small distortion after decoding. In actual facts, the low-rate/low-distortion accessibility is limited by theoretical arguments (Shannon theorem), and an optimal performance is then measured by the extent to which the Shannon bound is approached.

For media file compression, such as images, video, and music, this is often bearable. But, a more effective compression algorithm would side-step severe distortions. For certain data streams lossy compression can never be acceptable, of course.

The underlining structure of the protocol introduced in the present paper amounts to a CSP. Examples of CSPs are scheduling a group of jobs to carry out a more complex task, laying out the microscopic components of a silicon chip, interpreting a visual image, and of course, compressing a data string. The variables manifest in a CSP must all receive appropriate values within specified boundaries in order to solve the problem. The silicon chip for instance is not only bounded by the area of the device, but is also constrained by the total electrical requirements and heat production.

A CSP with N variables can be defined by a cost function and then defining a specific case by using actual variables. The function can be represented graphically so that one sees two kinds of nodes in the graph – the variables and the constraints. The problem is fully defined once we specify for which variable assignments the constraints are satisfied. In parity check gates, for example, a constraint is satisfied if the sum of the Boolean variables linked to it is even. The next stage is to demonstrate how a CSP can be used to compress data.

The team has defined a new kind of CSPs, using non-linear nodes. The satisfiability of one constraint does not depend on the sum of the variables but is a more complex function of them. Every bit of the string to

Fig. 1. a Factor graph for a CSP with $N = 7$ variables (circles) and $M = 5$ constraints (squares). In this example the connectivity of the function nodes is kept fixed, that is $K_a = K$ in the notations of the text. **b** Message passing procedure ($K = 6$): The probability $q_a \rightarrow o$ ($u_a \rightarrow o$) depends on all the probabilities $q_{b_i \rightarrow i}$ ($u_{b_i} \rightarrow i$), $i = 1, \dots, 5$ (see text).

1.D The Parity Source Coder

In order to give an example, we apply these ideas to the case where the CSP is a parity check (the optimization problem is then called XORSAT). In other words, the cost of the constraint a is written as

$$\varepsilon_a = 1 + (-1)^{x_a} \prod_{i \in a} \sigma_i . \quad (4)$$

This Parity Source Coder (PSC) is the counterpart of the LDPC codes in error-correcting codes and then it seems natural to expect a good performance from it.

In figure 2 we show that the ground state energy of the XORSAT problem, that is the theoretical capacity of the PSC, quickly approaches the Shannon bound as K increases [20]. A PSC with checks of degree $K \geq 6$ has then a theoretical capacity close to the optimal one. The problem here is that there does not exist any fast algorithm that can encode a string [for example, the survey inspired decimation (SID) algorithm that we discuss in the next section is known not to converge].

We will thus investigate a new kind of CSP, using non-linear nodes, with (almost) the same good theoretical performance as

the XORSAT problem, but with a fast encoding algorithm. Before we do this, we introduce in the next section the general formalism used in the study of non-linear nodes.

2 The General Formalism

Given a CSP at some α , we are interested in knowing whether a *typical* instance of the problem is satisfiable (i.e. all the constraints can be satisfied by one global configuration) as the number of variables goes to infinity. Generally speaking, there will be a phase transition between an SAT regime (at low α the problem has a small number of constraints and thus is solvable – at least in principle) and an UNSAT regime where there are too many constraints and one cannot find a zero-energy configuration. In this UNSAT regime, one wants to minimize the number of violated constraints. This kind of problem can be approached by message passing algorithms.

Due to the large length of typical loops, the local structure of a typical graph is equivalent to a tree, and this is crucial in what follows. We introduce the *cavity bias* $u_a \rightarrow i \in \{-1, 0, +1\}$ sent from a node a to

be compressed is used to control a check of this CSP. Then, we run an appropriated SP algorithm in order to find the global assignment of the variables which violate the minimum number of constraints. This is the encoded word. Decoding every single bit then amounts to looking at the variables which are attached to the corresponding check in the CSP and performing the non-linear operation.

This protocol gives the best performance for compression, both from the theoretical point of view (theoretical capacity close to the Shannon bound) and from the algorithmic perspective. In practice, it takes several hours to compress a string of $N = 1,000$ bits at a compression rate of $1/2$ using the team's general purpose software, although an improved specification for the computer code itself will improve performance significantly.

For example, the introduction of global structures incorporating the specific nature of the non-linear gates would lead to a reduction by a factor 10 or more in terms of CPU time, at the cost of some more memory usage which is still below the capacity of today's computers.

It is important to stress that while the protocol is extremely promising, the present status of its implementation is still rather primitive. A comparison with standard compression tools (as zip or jpeg) would first require a generalization of the method to deal with correlated and continuous sequences. The quantization step is then a crucial one, and the proper implementation of it is the last hurdle to be overcome before the algorithm becomes competitive at a non-academic level.

David Bradley of Sciencebase.com

Fig. 2. The ground state energy density for the XORSAT problem at $\alpha = 2.0$ vs. K . We also show the corresponding Shannon bound as computed from rate-distortion theory, $E_{Sh} = 2\alpha D_{Sh}$, where D_{Sh} is such that $1 - H_2(D_{Sh}) = 1/\alpha$.

a variable i . A non-zero message means that the variable i is requested to assume the actual value of $u_a \rightarrow i$ in order to satisfy the clause a . If $u_a \rightarrow i = 0$ the variable i is free to assume any value. It is clear that this message sent to i should encode the information that a receives from all the other variables attached to it. In order to clarify this point, we refer to figure 1b, that is we focus on a small portion of the graph. As the graph is locally tree-like, the variables σ_i , $i = 1, 2, \dots, K-1$, are only connected through clause a if N is large enough. If a is absent, the total energy of the system can be written as

$$E^N(\sigma_1, \dots, \sigma_{K-1}) = A - \sum_{i=1}^{K-1} h_i \sigma_i.$$

We have used here the assumption that the probability of these σ_i factorizes. This is again motivated by the local tree structure, but we shall come back to this point. After clause a is added, $E^{N+1}(\sigma_0, \sigma_1, \dots, \sigma_{K-1}) = E^N(\sigma_1, \dots, \sigma_{K-1}) + \varepsilon_a(\sigma_0, \sigma_1, \dots, \sigma_{K-1})$ and the variables rearrange in order to minimize the total cost. The minimization then defines $\tilde{\varepsilon}(\sigma_0)$ from

$$A - \sum_{i=1}^{K-1} |h_i| + \tilde{\varepsilon}(\sigma_0) = \min_{\sigma_1, \dots, \sigma_{K-1}} E^{N+1}(\sigma_0, \sigma_1, \dots, \sigma_{K-1}). \quad (5)$$

This $\tilde{\varepsilon}(\sigma_0)$ is then the cost to be paid for adding one variable with a fixed value σ_0 . Without losing generality, it can be written as

$$\tilde{\varepsilon}(\sigma_0) \equiv \Delta_{a \rightarrow 0} - \sigma_0 u_{a \rightarrow 0}, \quad (6)$$

where $u_{a \rightarrow 0}$ is the cavity bias acting on the new variable and $\Delta_{a \rightarrow 0}$ is related to the actual energy shift by

$$\Delta E \equiv \min_{\sigma_0, \dots, \sigma_{K-1}} [E^{N+1}(\sigma_0, \sigma_1, \dots, \sigma_{K-1}) - E^N(\sigma_1, \dots, \sigma_{K-1})] = \Delta_{a \rightarrow 0} - |u_{a \rightarrow 0}|. \quad (7)$$

Given that ε_a can be 0 or 2, depending on the set of fields $h_1, \dots, h_{K-1}$ one has four possibilities:

$$\begin{aligned} \tilde{\varepsilon}(+1) = 0 \quad \text{and} \quad \tilde{\varepsilon}(-1) = 0 &\Rightarrow u = 0, \quad \Delta = 0 \\ \tilde{\varepsilon}(+1) = 0 \quad \text{and} \quad \tilde{\varepsilon}(-1) = 2 &\Rightarrow u = +1, \quad \Delta = 1 \\ \tilde{\varepsilon}(+1) = 2 \quad \text{and} \quad \tilde{\varepsilon}(-1) = 0 &\Rightarrow u = -1, \quad \Delta = 1 \\ \tilde{\varepsilon}(+1) = 2 \quad \text{and} \quad \tilde{\varepsilon}(-1) = 2 &\Rightarrow u = 0, \quad \Delta = 2. \end{aligned} \quad (8, 9, 10, 11)$$

In other words, a non-zero message u is sent from clause a to variable σ_0 only if the satisfiability of clause a depends on σ_0 . A null message ($u = 0$) can occur in the two distinct cases (8) and (11).

The main hypothesis we have made so far consisted in assuming that the two variables are uncorrelated if they are distant (the energy is linear in the σ_i 's). It turns out that, in a large region of the parameter space [21, 22], including the regime we are interested in, this is not correct. This is due to the fact that the space of solutions breaks into many disconnected components if α is greater than a critical value. In order to deal with this case, one has to introduce, for each directed link, a probability distribution of the cavity biases, namely $q(u) \equiv \eta^+ \delta_{u,+1} + \eta^- \delta_{u,-1} + (1 - \eta^+ - \eta^-) \delta_{u,0}$. The hypothesis of no correlation holds if the phase space is restricted to one component. The interpretation of $q_a \rightarrow i(u_a \rightarrow i)$ is the probability that a cavity bias $u_a \rightarrow i$ is sent from clause a to variable i when one component is picked at random [23]. According to the rules (8, 9, 10, 11), and with the topology of figure 1b as a reference for notations, the survey propagation (SP [24]) equations are shown in equations 12, 13, 14, where the energy shift ΔE is given in equation 7 and is non-zero only when the constraint is UNSAT for any value of σ_0 (see equation 11). The crucial reweighting

term $\exp(-y\Delta E)$ thus acts as a 'penalty' factor each time a clause cannot be satisfied. This term is necessary in the UNSAT regime which we explore here (while simpler equations with $y = \infty$ are enough to study the SAT phase).

2.A Encoding: SP and Decimation

For each fixed value of y , the iterative solution for the SP equations can be implemented on a single sample [24], i.e. on a given graph where we know all the function nodes involved. The cavity probability distributions $q(u)$'s are updated by picking up one edge at random and using (8, 9, 10, 11). This procedure is iterated until convergence. This yields a set of messages $\{\eta_{a \rightarrow i}^-, \eta_{a \rightarrow i}^0, \eta_{a \rightarrow i}^+\}$ on each edge of the factor graph which is the solution of the SP equations. This solution provides very useful information about the single instance that can be used for decimation. As explained [25], the finite value of y used for this purpose must be properly chosen. Given the solution for this y , one can compute the distribution $P(H_i)$ of the total bias $H_i = \sum_{a \in i} u_{a \rightarrow i}$ on each variable. One can then fix the most biased variables, i.e. the one with the largest $|P(H_i = +1) - P(H_i = -1)|$, to the value suggested by the $P(H_i)$ itself. This leads to a reduced problem with $N - 1$ variables. After solving again the SP equations for the reduced problem, the new most-biased variable is fixed and one carries on until the problem is reduced to an 'easy' instance. This can be finally solved by some conventional heuristic (e.g. walksat or simulated annealing). A significant improvement of the decimation performance can be obtained by using a backtracking

$$\begin{aligned} \eta_{a \rightarrow 0}^+ &\propto \text{Prob} \left[\left\{ (u_{b \rightarrow i_1})_{b \in i_1 \setminus a}, \dots, (u_{b \rightarrow i_{K-1}})_{b \in i_{K-1} \setminus a} \right\} \mid (\tilde{\varepsilon}_a(+1) < \tilde{\varepsilon}_a(-1)) \right] e^{-y\Delta E}, \\ \eta_{a \rightarrow 0}^- &\propto \text{Prob} \left[\left\{ (u_{b \rightarrow i_1})_{b \in i_1 \setminus a}, \dots, (u_{b \rightarrow i_{K-1}})_{b \in i_{K-1} \setminus a} \right\} \mid (\tilde{\varepsilon}_a(+1) > \tilde{\varepsilon}_a(-1)) \right] e^{-y\Delta E}, \\ \eta_{a \rightarrow 0}^0 &\propto \text{Prob} \left[\left\{ (u_{b \rightarrow i_1})_{b \in i_1 \setminus a}, \dots, (u_{b \rightarrow i_{K-1}})_{b \in i_{K-1} \setminus a} \right\} \mid (\tilde{\varepsilon}_a(+1) = \tilde{\varepsilon}_a(-1)) \right] e^{-y\Delta E}, \end{aligned} \quad (12, 13, 14)$$

procedure [25, 26]: at each step we also rank the fixed variables with a strongly opposed bias and unfix the most ‘unstable’ variable with finite probability. The algorithm described here is called SID and its peculiar versions have recently been shown to be very useful in many CSP problems. Unfortunately, the basic version of SID does not work for the XORSAT problem because of the symmetric character of the function nodes that is reflected in a large number of unbiased variables [some improvements appear to be possible; Wainwright, pers. commun.].

2.B Statistical Analysis: Theoretical Performance

One can perform a statistical analysis of the solutions of the SP equations by population dynamics [23]. The knowledge of the function node ε_a allows to build up a table of values of u for each configuration of the local fields $h_1, \dots, h_{K-1}$ (this is done according to the minimization procedure described above). We then start with some initial (random) population of $\vec{\eta}_i \equiv \{\eta_i^+, \eta_i^0, \eta_i^-\}$, for $i = 1, \dots, N$. We extract a Poisson number p of neighbours and p probabilities $\vec{\eta}_{i_1}, \dots, \vec{\eta}_{i_p}$. According to these weights, p biases $u_1, \dots, u_p$ are generated and their sum computed, $h_i = \sum_{a=1}^p u_a$. Once we have $K-1$ of these fields we perform the minimization in equation 5 and compute the new probability for u_a according to the rules (8,9,10,11). The whole process is then iterated until a stationary distribution of the cavity biases is reached. This method for solving the SP equations is very flexible with respect to the change of the choice of the node, because this choice just enters into the calculation once at the beginning of the algorithm in order to initialize some tables. One can also study problems with many different types of nodes in a given problem: in this case, one of them is randomly chosen each time the updating is performed. We finally stress that once the probability distribution of the cavity bias-

es is known the ground state energy of the problem can be computed according to the formalism introduced in Mézard and Zecchina [23]. The expression for the energy in terms of the probability distributions is shown in equations 15,16,17, the overline representing an average over the Poisson distribution and the choice of the distributions $q_1, \dots, q_p$ in the population. Recalling that $R = 1/\alpha$, the average distortion (i.e., the theoretical capacity) of a compressor based on this CSP is computed through equations 3 and 15.

Let us study here as an example a family of function nodes whose energy is fully invariant under permutations of the arguments. These nodes can be classified according to the energy of the node for a given value of the ‘magnetization’ $m \equiv \sum_{i=1}^K \sigma_i$. We restrict to odd K and label a particular node with the sequence of values $\{\varepsilon(m=K), \varepsilon(m=K-2), \dots, \varepsilon(m=1)\}$. In figure 3 we report the results of the population dynamics algorithm for some types of nodes at $K=7$. According to equation 15, the ground state energy $E(\alpha)$ corresponds to the maximum over y of the

$$E(\alpha) = \max_y \Phi(\alpha, y), \quad (15)$$

$$\Phi(\alpha, y) = \frac{(-1)}{y} \left\{ [1 + (K-1)\alpha] \overline{\log A_p(y)} - (K-1)\alpha \overline{\log A_{p+1}(y)} \right\}, \quad (16)$$

$$A_p(y) \equiv \sum_{u_1, \dots, u_p} q_1(u_1) \cdots q_p(u_p) \exp \left(y \left| \sum_{a=1}^p u_a \right| - y \sum_{a=1}^p |u_a| \right), \quad (17)$$

Fig. 3. **a** The free energy $\Phi(\alpha, \gamma)$ of a number of symmetric nodes with $K=7$ at $\alpha=1.4$, classified according to the rule given in the text. The asymmetry $|\eta_+ - \eta_-|$ of these nodes **(b)** is measured as well as the ‘paramagnetic degree’ $\eta_0 = 1 - \eta_+ - \eta_-$ **(c)**. Note that the usual 7-XOR node corresponds to the $\{2, 0, 2, 0\}$ case in this notation.

free energy $\Phi(\alpha, y)$ represented in figure 3a. The theoretical performance of these nodes is quite close to the PSC case (the XOR node characterized by the sequence $\{2, 0, 2, 0\}$).

3 Non-Linear Nodes

We consider in this section the function nodes that give the best performance for compression, both from the theoretical point of view (theoretical capacity close to the parity-check nodes) and from the algorithmic aspect (the SID algorithm at finite y is found to converge in the UNSAT regime, thus giving an explicit encoding algorithm). These non-linear function nodes are defined as follows. We recall that the output ε_{XOR} of a K -XORSAT node is given by twice the sum modulo 2 of the input bits. We label each configuration $\vec{\sigma} \in \{0, 1\}^K$ with an integer $l \in [1, 2^K]$ and consider a random permutation π of the vector $\{1, 2, \dots, 2^K\}$. Then, we can associate the output of the *random node* by letting $\varepsilon^{(\pi)}(l) = \varepsilon_{\text{XOR}}(\pi(l))$. In this way we are left with a random but balanced output which can be different from XOR. Also, it is

clear that they are not more defined by a linear formula over the Boolean variables.

We can take advantage of the formalism introduced in the previous section in order to study the theoretical performance of these new function nodes. In particular, we have used 10–30 different random nodes to build the factor graphs. In order to improve the performance, we have also not allowed ‘fully canalizing’ nodes, that is nodes whose SAT character depends on just one variable.

In figure 4 we show our results. The ground state energy is shown to quickly approach the Shannon bound as K increases and for any α . As an example, at $\alpha = 2$ (corresponding to a compression rate $R = 1/2$) the difference between the $K = 8$ value and the theoretical limit is $\approx 2\%$. This looks very promising from the point of view of data compression.

Furthermore, the results obtained from the SID (same plot) show that in this case the algorithm does converge and its performance is very good. It should be noted that when K becomes large the difference between SP and the belief propagation

(BP) becomes small (this can be seen for instance in the analysis of Ciliberti [20]); in fact BP does also provide a good encoding algorithm for $K \gtrsim 6$. At fixed K , the time needed to solve the SP equations is $\mathcal{O}(N \log N)$. The actual computational time required by the decimation process is of the same order and it slightly depends on the details of the SID algorithm (e.g. the number of variables fixed at each iteration, whether or not a backtracking procedure is used, which kind of heuristic is adopted to solve an ‘easy’ reduced instance, the proper definition of the latter, etc.). The dependence on K is exponential. Thus, even if increasing K is good from a theoretical point of view, it turns out to be very difficult to work at high K . In practice, it takes a few hours to compress a string of $N = 1,000$ bits at $K = 6$ by using our general purpose software. We think that some more specified codes would lead to a better performance.

To conclude, we have shown how the methods of statistical mechanics, properly adapted to deal with a new class of CSP, make it possible to implement a new protocol for data compression. The new tool introduced here, the non-linear gates, looks very promising for other practical applications in information theory.

Acknowledgements

We thank A. Montanari, F. Ricci-Tersenghi, D. Saad, M. Wainwright and J.S. Yedidia for interesting discussions. S.C. is supported by the EC through the network MTR 2002-00307, DYGLAGEMEM. This work was supported in part by the EC through the network MTR 2002-00319 STIPCO and the FP6 IST consortium EVERGROW.

Fig. 4. The ground state energy for a system with 30 non-linear nodes. We also plot the Shannon bound and the performance of the SID algorithm for the $K = 6$ case.

References

- 1 Gallager RG: Low Density Parity-Check Codes. Cambridge, MIT Press, 1963.
- 2 Spielman DA: The complexity of error-correcting codes. *Lecture Notes in Computer Science* 1997; 1279: 67–84.
- 3 MacKay D: *Information Theory, Inference, and Learning Algorithms*. London, Cambridge University Press, 2003.
- 4 Sourlas N: Spin-glass models as error-correcting codes. *Nature* 1989; 339: 693–694.
- 5 Montanari A: The glassy phase of Gallager codes. *Eur Phys J* 2001; 23: 121–136.
- 6 Nishimori H: *Statistical Physics of Spin Glasses and Information Processing: an Introduction*. Oxford, Oxford University Press, 2001.
- 7 Jelinek F: Tree encoding of memoryless time-discrete sources with a fidelity criterion. *IEEE Trans Inform Theory* 1969; 15/6: 584.
- 8 Martinian E, Yedidia JS: MERL report, 2003.
- 9 Ziv J, Lempel A: A universal algorithm for sequential data compression. *IEEE Trans Inform Theory* 1977; 23: 337.
- 10 Cleary J, Witten I: Data compression using adaptive coding and partial matching. *IEEE Trans Commun* 1984; 32: 396.
- 11 Willems F, Shtarkov Y, Tjalkens T: The context-tree weighting method: Basic properties. *IEEE Trans Inform Theory* 1995; 41: 653.
- 12 Berger T, Gibson JD: Lossy source coding. *IEEE Trans Inform Theory* 1998; 44/6: 2693.
- 13 Ciliberti S, Mézard M, Zecchina R: Lossy data compression with random gates. *Phys Rev Lett* 2005; 95: 038701.
- 14 Hosaka T, Kabashima Y, Nishimori H: Statistical mechanics of lossy data compression using a non-monotonic perceptron. *Phys Rev E* 2002; 66: 066126.
- 15 Battaglia D, Braunstein A, Chavas J, Zecchina R: Source coding by efficient selection of ground states clusters. *Phys Rev E* 2005; 72: 015103.
- 16 Garcia-Frias J, Zhao Y: Compression of binary memoryless sources using punctured turbo codes. *IEEE Commun Lett* 2002; 394.
- 17 Caire G, Shamai S, Verdù S: A new data compression algorithm for sources with memory based on error correcting; in *Proc IEEE Inform Theory Workshop*, Paris, 2003, p 291.
- 18 Matsunaga Y, Yamamoto H: A coding theorem for lossy data compression by LDPC codes; in *Proc IEEE Int Symp Inform Theory*, Lausanne, 2002, p 461.
- 19 Gray RM, Neuhoff DL: Quantization. *IEEE Trans Inform Theory* 1998; 44: 2325.
- 20 Ciliberti S, Mézard M: The theoretical capacity of the Parity Source Codes. *J Stat Mech: Theory and Experiments* 2005; P10003
- 21 Mézard M, Parisi G, Zecchina R: Analytic and algorithmic solution of random satisfiability problems. *Science* 2002; 297: 812.
- 22 Biroli G, Monasson R, Weigt M: A variational description of the ground state structure in random satisfiability problems. *Eur Phys J* 2000; B14: 551.
- 23 Mézard M, Zecchina R: The random K-satisfiability problem: From an analytic solution to an efficient algorithm. *Phys Rev* 2002; E66: 056126.
- 24 Braunstein A, Mézard M, Zecchina R: Survey propagation: an algorithm for satisfiability. *Random Structures Algorithms* 2005; 27: 201–226.
- 25 Battaglia D, Kolár M, Zecchina R: Minimizing energy below the glass thresholds. *Phys Rev* 2004; E70: 036107.
- 26 Parisi G: A backtracking survey propagation algorithm for K-satisfiability. *cond-mat/0308510*.