Compression by replication

Roberto C. Alamino, Juan P. Neirotti, and David Saad
Non-linearity and Complexity Research Group, Aston University, Birmingham B4 7ET, United Kingdom
(Received 11 September 2013; revised manuscript received 24 January 2014; published 4 March 2014)

A recently introduced inference method based on system replication and an online message passing algorithm is employed to complete a previously suggested compression scheme based on a nonlinear perceptron. The algorithm is shown to approach the information theoretical bounds for compression as the number of replicated systems increases, offering superior performance compared to basic message passing algorithms. In addition, the suggested method does not require fine-tuning of parameters or other complementing heuristic techniques, such as the introduction of inertia terms, to improve convergence rates to nontrivial results.

DOI: 10.1103/PhysRevE.89.033301 PACS number(s): 05.10.−a, 84.35.+i, 89.70.−a

I. INTRODUCTION

The successful application of techniques developed in statistical mechanics of disordered systems to a wide range of problems in information theory has benefited both fields by exchanging methods and ideas, providing new insights and algorithmic tools [1]. The statistical mechanics methodology has complemented the mathematical rigor of traditional information theory techniques by providing exact analytical results for typical properties in the limit of very large systems—the thermodynamic limit. Conversely, the importance of obtaining *specific* microscopic states in practical information theory problems, in contrast to the usual goal of characterizing macroscopic states in statistical mechanics, has contributed to a renewed interest in inference methods to obtain ground-state solutions of the corresponding Hamiltonians when the energy landscape is complex.

In problems related to communication systems, microscopic states are associated with specific transmitted messages and one is interested in recovering messages associated with the specific instance rather than analyzing general macroscopic properties averaged over an ensemble of such systems. The correct message refers to the ground state of the corresponding Hamiltonian with the level of noise in the channel being represented by the temperature used. Common examples are error correcting codes [2], where one wishes to recover the original message, after it has been encoded and corrupted upon transmission through a noisy channel.

The problem of finding a ground state, or equivalently the global minimum of a Hamiltonian, can only be solved analytically in very simple cases. For disordered systems, especially in the spin-glass phase, the energy landscape is so complex that the use of approximate computational techniques is unavoidable. The ruggedness of energy landscapes which characterize such systems poses a challenge for optimization techniques. For instance, gradient descent-based methods get trapped in local minima and more sophisticated Monte Carlo algorithms, such as parallel tempering [3,4], are computationally slow.

An alternative family of algorithms which provide computationally efficient approximations to the exact but computationally hard full Bayesian inference is that of message passing (MP) algorithms, also known as belief propagation [5]. These methods have been able to achieve good performance in many complex problems and are considered a promising alternative for tackling inference problems in a range of fields such as information theory [1], hard combinatorial problems [6], statistical mechanics models, and complex systems in general.

The information theoretical problem we address here is that of source coding or lossy compression. The problem is of great importance practically and is highly challenging theoretically; computationally efficient solutions for this problem have been sought after for over 50 years. Shannon was the first to study lossless and lossy [7,8] compression and to establish theoretical bounds to the achievable performance under a given information loss. However, Shannon's results are not constructive, leaving open the challenge of finding a computationally feasible scheme that saturates the theoretical bounds.

The main difficulty in finding such schemes is the associated computational complexity. Some schemes can saturate the theoretical bounds, for instance by an exhaustive search, but are impractical due to the computational cost involved which scales exponentially with the size of the message. Other approaches provided good approximations [9–14] that still fall short of the theoretical limits for certain loss rates. The search for efficient schemes, those which are at least polynomial in message size, is what drives research in the field even today. Notable among these schemes are recent approximate Bayesian methods based on MP algorithms.

A radically different approach based on the nonlinear perceptron has been introduced by Hosaka, Kabashima, and Nishimori [15]. By using the replica method it has been shown that a nonlinear perceptron can be used as part of a compression scheme, which can achieve close to optimal performance, both in terms of the theoretical compression-distortion limits [15] and the related error exponents [16], depending on the parameters of the message generation and activation function of the perceptron. The analytical results are obtained for the typical case and were numerically verified only by exhaustive search methods, which are clearly exponentially slow as the size of the message is increased. In a follow-up work [17] an MP algorithm has been suggested for the compression of the messages showing good performance as long as some heuristic modifications were added; but performance was bounded due to inherent limitations of the inference method. While this compression method is highly promising it still requires an efficient inference technique to bring performance closer to the theoretical limits.

The aim of the current paper is to do exactly that; namely, to adapt a recently introduced MP algorithm [18] to bring

the performance of the nonlinear perceptron compression scheme closer to the theoretical limits. This algorithm, named the replicated online message passing algorithm (rOnMP), is based on improved inference in hard computational problems by averaging over results obtained from different solutions of replicated variable systems. It has been previously applied to solve the binary Ising perceptron capacity problem as a benchmark case.

We will employ a similar replicated online message passing algorithm to obtain solutions as part of the compression method and show that it can achieve an increasingly better performance as the number of replica increases. The rOnMP algorithm exhibits several advantages over basic MP methods: (i) it does not require any heuristic additions to suppress convergence to nontrivial solutions due to symmetry properties of the problem; (ii) it does not depend on training parameters that may need fine-tuning; (iii) and, most importantly, it provides increasingly improved performance as the number of replica increases.

The remainder of the paper is organized as follows. In Sec. II we provide the theoretical formulation of the compression method using a nonlinear perceptron. We then proceed to introduce the conventional MP equations corresponding to this method in Sec. III, which use the whole data set at once (offline). In Sec. IV we argue in favor of converting these equations into an online set of equations that incorporate a single data point at a time. A further improvement that completes the algorithm is the replication of the online MP process, as explained in Sec. V. Once the rOnMP algorithm has been fully derived for the compression problem, we analyze its computational complexity in Sec. VI. Results of numerical simulations are presented in Sec. VII followed by a summary and some final considerations given in Sec. VIII.

II. COMPRESSION BY A NONLINEAR PERCEPTRON

The compression problem consists in encoding an N-dimensional binary message $\mathbf{y} = (y_1, \ldots, y_N) \in \{\pm 1\}^N$ into a K-dimensional binary vector $\mathbf{b} = (b_1, \ldots, b_K) \in \{\pm 1\}^K$, where $K < N$, such that the compressed message can be later recovered by a decompressing algorithm with zero or minimal loss. The compression rate $R = K/N$ indicates the level of compression; it is desirable to minimize R while minimizing distortion losses. When zero information loss is possible in the recovered message we term the problem lossless compression, while when allowing for some deterioration after recovery it is called lossy compression.

Given that we will usually use finite compression rates in this study, we will refer to both N and K interchangeably as the *system size*. The thermodynamic limit will then be taken by sending both N and K to infinity while keeping R fixed to a finite value.

Shannon's source-code theorem [7] shows that lossless compression is possible when the rate R is less than the entropy per bit of the source $\mathbf{y}$ in the thermodynamic limit. Higher compression rates can be achieved if one allows for information loss, with the precise meaning that a nonvanishing average error per bit will be expected in the retrieved message. The error per bit, also called the *distortion rate*, is measured by the average Hamming distance between the original message $\mathbf{y}$ and its inferred version $\hat{\mathbf{y}}$ as

$$D = \lim_{N\to\infty} \frac{1}{N} \sum_{\mu=1}^{N} \delta(y_\mu, \hat{y}_\mu), \tag{1}$$

where δ represents the Kroenecker δ.

Perceptrons represent simple nonlinear maps and have been extensively studied in statistical mechanics [19]. As such, they are promising candidates for compression schemes; one such specific scheme was proposed in [15]. The perceptron used corresponds to the mapping

$$y_\mu = \mathrm{sgn}\,(\Delta - |\xi_\mu|), \tag{2}$$

with the so-called *synaptic field* given by

$$\xi_\mu = \frac{1}{\sqrt{K}} \sum_{k=1}^{K} b_k s_{\mu k}, \tag{3}$$

where the *a priori* given vectors $\{\boldsymbol{s}_\mu\}$ are fixed at each instance of the problem. The nonlinear activation function used is visualized in Fig. 1, where it can be seen that the constant threshold Δ defines the width of the square bump.

In the suggested compression scheme, a fixed set of N randomly sampled K-dimensional vectors $\boldsymbol{s}_\mu \in \mathbb{R}^K$ is given, playing the role of fixed parameters that characterize the compression scheme and facilitate decompression. The data set $\mathcal{D}$ composed of pairs $(y_\mu, \boldsymbol{s}_\mu)$ is used to estimate the parameters of the perceptron, which represent the compressed codeword corresponding to the original data vector $\mathbf{y}$. These parameters are encoded by the vector $\mathbf{b}$, known in the literature as the *synaptic vector*. Decompression consists in presenting the input vectors $\boldsymbol{s}_\mu$ to the perceptron to obtain the decompressed message $\hat{\mathbf{y}}$ using Eq. (2).

Typical case analysis of the achievable compression rate for a given distortion (error rate) D was carried out using the replica method, with replica symmetry being sufficient in this case [15]. The data bits $y_\mu \in \{\pm 1\}$ of the original message were randomly sampled from a biased distribution $p = \mathcal{P}(y_\mu = 1) = 1 - \mathcal{P}(y_\mu = -1)$; the parameter vectors $s_{\mu k}$ were sampled from a Gaussian distribution of zero mean and unit variance. Theoretically, the compression scheme

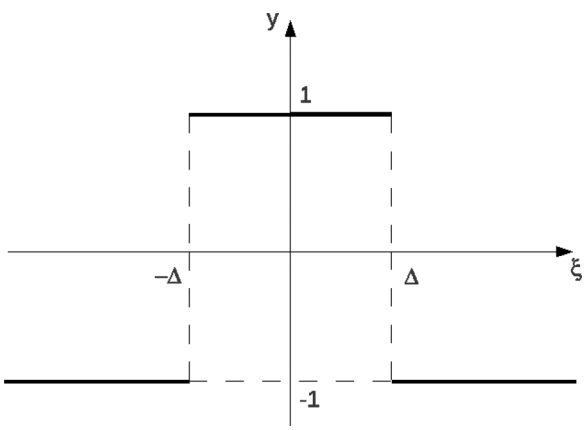

FIG. 1. Activation function for the nonlinear perceptron used in the present compression scheme.

saturates Shannon's limit, with a compression rate $R(D)$

$$R = H_2(p) - H_2(D) \tag{4}$$

and

$$\mathrm{erf}\left(\frac{\Delta}{\sqrt{2}}\right) = \frac{p-D}{1-2D}, \tag{5}$$

where $H_2(x) = -x\log_2 x - (1-x)\log_2(1-x)$.

Equation (4) represents the optimal compression rates for a given fixed D for any message bias, while Eq. (5) gives the optimal value of Δ in terms of the bias and either D or R. We will use these results to determine Δ in our scheme and compare the performance of our algorithm with optimal compression.

III. MESSAGE PASSING

The validity of the replica solution for the suggested compression scheme was tested in [15] using an exhaustive search, which is very slow and practically infeasible as it scales exponentially with the system size. An MP algorithm aimed at implementing the compression scheme was subsequently suggested in [17]. Our objective in this paper is to suggest an alternative efficient compression algorithm, where "efficient" refers to the algorithm's computational complexity which scales polynomially with the system size.

The algorithm we propose here is based on a recently introduced variant of the MP algorithm [18]. The latter aims at addressing one particularly serious recurring general problem in perceptron optimization tasks—the complexity of the energy landscape.

Two key modifications of traditional MP algorithms were fundamental to improving the quality of the solutions obtained.

(1) Making MP an online algorithm. The conventional MP algorithm, which aims to provide a good approximation to the maximum *a posteriori* solution for all available data simultaneously, is an offline process expressed in the form of recursive equations. We modified these equations to devise an online algorithm where data points (input patterns and the corresponding output binary values) are introduced one at a time. This allows one to explore a new degree of freedom which does not exist in the offline version, namely the order of data presentation.

(2) Replicating the process. We introduced a series of real replicated systems exposed to the same set of data and constraints, but setting a different path through example space for each of the systems. Final estimates are obtained by averaging over the inferred solutions calculated by each one of the replicated systems.

The method proved to be extremely good in tackling the binary Ising perceptron capacity problem, which is computationally hard in both worst and typical cases. One of the strengths of this method is in its generality; in principle, it can be easily applied to any nonpathological densely connected inference problem with minimal modifications. At the heart of the method, which we abbreviated by rOnMP (replicated online MP), are the original MP equations. Because we are dealing with a densely connected system, we need to derive an approximation for these equations. This is done by an expansion which is valid in the limit of large systems. Details of this approximation and the following derivation were given in [18] and could easily be adapted for the current case by introducing straightforward modifications. We will therefore provide here only a brief description of the derivation and refer the reader to [18] for further details.

The ordinary MP equations are given as pairs of coupled equations for each cavity magnetization m_μ. These equations take the form

$$\hat{m}^t_{\mu k} = \frac{\sum_{b_k} b_k \mathcal{P}^{t+1}(y_\mu|b_k,\{y_{\nu\neq\mu}\})}{\sum_{b_k} \mathcal{P}^{t+1}(y_\mu|b_k,\{y_{\nu\neq\mu}\})}, \tag{6}$$

$$m^t_{\mu k} = \tanh\left[\sum_{\nu\neq\mu} \mathrm{arctanh}\,\hat{m}^t_{\nu k}\right] \approx \tanh\left(\sum_{\nu\neq\mu}\hat{m}^t_{\nu k}\right). \tag{7}$$

The temporal index t in the variables indicates the update order as the equations should be solved by iteration until they converge. The variables $\hat{m}$ are auxiliary variables used to calculate the actual cavity magnetizations m and are sometimes called *conjugate magnetizations*.

The method proposed in [18] can then be easily adapted to the compression scheme. Using the nonlinear activation function for the current perceptron, one can calculate analytically both the numerator and denominator of the equation for the conjugate magnetizations as a power series in K. This is accomplished by uncoupling the synaptic vectors by means of Hubbard-Stratonovich fields, which can then be exactly integrated at the end. By expanding the solution to leading order in K we finally obtain

$$\hat{m}_{\mu k} = \frac{2s_{\mu k}y_\mu}{\sqrt{K}} \frac{\mathcal{N}\left(\Delta|-u_{\mu k},\sigma^2_{\mu k}\right) - \mathcal{N}\left(\Delta|u_{\mu k},\sigma^2_{\mu k}\right)}{1 + y_\mu\left[\mathrm{erf}\left(\frac{\Delta+u_{\mu k}}{\sqrt{2\sigma^2_{\mu k}}}\right) + \mathrm{erf}\left(\frac{\Delta-u_{\mu k}}{\sqrt{2\sigma^2_{\mu k}}}\right) - 1\right]}, \tag{8}$$

where

$$\mathcal{N}(x|u,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left[-\frac{(x-u)^2}{2\sigma^2}\right], \tag{9}$$

$$\sigma^2_{\mu k} = \frac{1}{K}\sum_{l\neq k}\left(1-m^2_{\mu l}\right)s^2_{\mu l}, \tag{10}$$

$$u_{\mu k} = \frac{1}{\sqrt{K}}\sum_{l\neq k} m_{\mu l}s_{\mu l}, \tag{11}$$

and $\mathrm{erf}(x)$ is the error function given by

$$\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}}\int_0^x d\tau\, e^{-\tau^2}. \tag{12}$$

The new pair of equations to be iterated until convergence are now (7) and (8). By dropping the index t to indicate fixed-point values once convergence is attained, we can write the inferred value of the perceptron synaptic vector b_k as

$$b_k = \mathrm{sgn}\, m_k, \quad m_k = \tanh\left(\sum_\mu \hat{m}_{\mu k}\right). \tag{13}$$

IV. OFFLINE VS ONLINE

The equations derived in the previous section are the conventional offline version of the MP algorithm, which means

that the data set used during the learning phase is available in full to the algorithm. The algorithm then uses the whole data set to extract the necessary information for the inference task. While probabilistically this is clearly optimal, it is unclear that the dynamical iterative process of Eqs. (6) and (7) will indeed converge to the correct solution in general graphs. In some cases, making use of the full data set sets the iteration dynamics on a course that prevents convergence to the optimal solution.

Equations (6), (7), and (13) are fairly general offline equations for binary systems and the (nonreplicated) MP approach of [17] is, in practice, equivalent to them. However, this set of equations has an inconvenient symmetry, which gives rise to an ambiguity in deciding on the sign of the inferred variables. The reason is that, due to this symmetry, the equations result in $m_k = 0$. The equations introduced in [17], as the ones derived here, consider the first significant term in $\hat{m}_{\mu k}$ and hence suffer from a similar symmetry problem. To solve it, a heuristic inertia term was introduced [20], which depends on a parameter that has to be adjusted by trial and error.

Given our success in solving the Ising perceptron capacity problem by turning the offline MP into an online algorithm we introduce a similar approach here. The online version of the MP equations is obtained via an additional expansion for the large system size, this time using Eq. (13).

Because in this equation each term of the summation inside the hyperbolic tangent is of order $1/\sqrt{K}$, we can single out one of these terms and expand the tanh around the remaining terms for large K. By singling out the νth term, the right-hand side of Eq. (13) becomes

$$m_k = \sum_{n=0}^{\infty} \frac{\hat{m}_{\nu k}^n}{n!} F_n(m_{\nu k}), \tag{14}$$

where

$$m_{\nu k} = \tanh\left(\sum_{\mu \neq \nu} \hat{m}_{\mu k}\right) \tag{15}$$

and

$$F_n(m_{\nu k}) = \frac{d}{dx} \tanh x \Big|_{x=\sum_\mu \hat{m}_{\mu k}}. \tag{16}$$

The fact that each F_n depends only on $m_{\nu k}$ is a consequence of a property of the tanh function, the derivative of the hyperbolic tangent being a function of the hyperbolic tangent itself.

We now reinterpret the term which was singled out as a new example, introduced after the previous patterns have already been learned. Alternatively, one may interpret the index ν in the MP equations as a time step t and substitute $m_{\nu k}^t$ by $m_k(t-1)$ and m_k^t by $m_k(t)$. We ran an extensive series of tests that led to the conclusion that by using an expansion up to the third term in Eq. (14) one can avoid the problematic symmetry effects of the offline MP and obtain extremely good compression performance. More explicitly, the expansion gives the following update rule:

$$\begin{aligned} m_k(t) = m_k(t-1) &+ \left[1 - m_k^2(t-1)\right]\hat{m}_{tk} \\ &- m_k(t-1)\left[1 - m_k^2(t-1)\right]\hat{m}_{tk}^2. \end{aligned} \tag{17}$$

At first sight, it seems that the second-order term would contribute to further stabilization of the zero solution as it suppresses the previous value of m_k. However, due to the fact that the first-order term gives rise to a solution that is identically zero, the second-order term actually generates a perturbation away from zero; while this perturbation is small, it helps break the symmetry enough to allow the algorithm to pick a sign for the magnetization.

The conversion of the offline algorithm to an online one is very important as it opens up the possibility for using a degree of freedom which was not available before—the order of data presentation. As we have seen in the beginning of this section, an offline algorithm like MP has access to the whole data set, which determines the course of the iteration dynamics. Conversely, online algorithms have access only to the information available *before* some point in time. Inference is then updated for each time step using new data in the order it arrives. This adds disorder to the iterative equations and prevents a set course for their dynamics; this randomness helps to avoid getting trapped in dynamical local minima, a fundamental limitation of the offline MP in some instances.

V. REPLICATION

The final and most important step of the rOnMP algorithm is the system replication. Replication, in the context of this algorithm, means the introduction of real replica of the same system, which carry out the same inference task in parallel (subject to the disorder in the introduction of examples) and interact at very specific points.

This replication is done by generating randomly n different paths in the example space. Each path is composed by the same example pairs $(y_\mu, \boldsymbol{s}_\mu)$, but in a different order. The idea, as we have already discussed briefly, is to use the new degree of freedom encoded by the order of example presentation to facilitate a better search in solution space and avoid being trapped in suboptimal minima. For N examples, there are $N!$ possible orders of presentation, but we will choose only a number n of these sequences, with n being of polynomial order in N. In previous applications [18] we observed that this is enough to considerably improve the performance of the nonreplicated algorithm.

Once n different paths through the example space have been generated, parallelization takes place. For each example path, a different replicon of the system is created. Each replicon works as an independent system performing online MP learning after each corresponding example is presented, and inferring a new $\boldsymbol{b}^a$, where a is a replica index ranging from 1 to n, and the components of each vector are given by Eq. (13) applied to each replicon.

After each one of these learning steps, an averaged inferred vector is calculated by taking a weighted average of all replica as

$$\bar{b}_k = \operatorname{sgn}\left(\sum_{a=1}^{n} w^a b_k^a\right), \tag{18}$$

and this is used as the initial point for the next learning step for each replicon.

The crucial point in the rOnMP algorithm is clearly how to decide on the appropriate weights for averaging. Although white averages, with all weights equal to 1, are usually faster to calculate, they exhibit very poor performance in the present case. One can alternatively adopt a procedure based on a Boltzmann weight

$$w^a \propto e^{-\beta E(\boldsymbol{b}^a)}, \tag{19}$$

with the energies $E(\boldsymbol{b}^a)$ being a measure of performance, which here we define as the number of misclassified examples. The parameter β works as an inverse temperature and we recover the white average for $\beta = 0$ (infinite temperature).

For the compression task, we observed that much better performance is attained when the average is highly biased, which is equivalent to choosing a very low temperature, thus selecting lower-energy states. We adopted the rather simplified criteria of choosing the best replicon as the inferred vector for the next learning step, which amounts to choosing the zero-temperature weights.

VI. COMPUTATIONAL COMPLEXITY

There is of course a trade-off between performance and computational complexity of the algorithm which cannot be avoided in most hard computational problems and the current one is no exception. The complexity of the energy landscape for the present problem suggests that exact algorithms are invariably computationally hard. The more sophisticated the search algorithm is, the more one can expect the computational complexity to increase. However, given the difficulty of the task, as long as the complexity of the resulting algorithm remains polynomial in the system size with a small power, this can be considered an acceptable trade-off.

As we are assuming that the system size K scales with the size of the data set N, the naive MP algorithm, summarized in Eqs. (6) and (7), requires the calculation of $2K^2$ terms as each equation depends on two indices. To calculate the hatted variables, Eq. (8) requires two loops, each one of order K. Calculating the magnetizations in Eqs. (10) and (11) also requires one internal loop of order K, making the total number of operations scale with K^3. The final step of the inference algorithm (13) does not increase the complexity as it requires only K^2 operations.

Although replication increases the computational complexity of the original MP equations, because we work with at most an order N of replica, this is not excessive given the computational complexity of the nonreplicated algorithm. The replicated algorithm scales with K^4, which is still polynomial in the system size, one order higher than the nonreplicated algorithm. The replica averaging operation does not change this result as it scales only with K. Therefore, the computational complexity of the rOnMP algorithm remains polynomial in the number of examples N as one would aim for in the case of hard computational problems.

It is not difficult to see that there is additional inherent complexity in the algorithm, hidden in the procedure for deciding on the order of example presentations. The most efficient way of choosing this order, which maximizes the information gain, is indeed a difficult problem and needs to be considered with much more detail. In our tests, we used a random order for the introduction of examples. Although this is far from optimal, even this very naive approach resulted in a considerable improvement in the algorithm's performance. This suggests that by improving this procedure, one can achieve even better results.

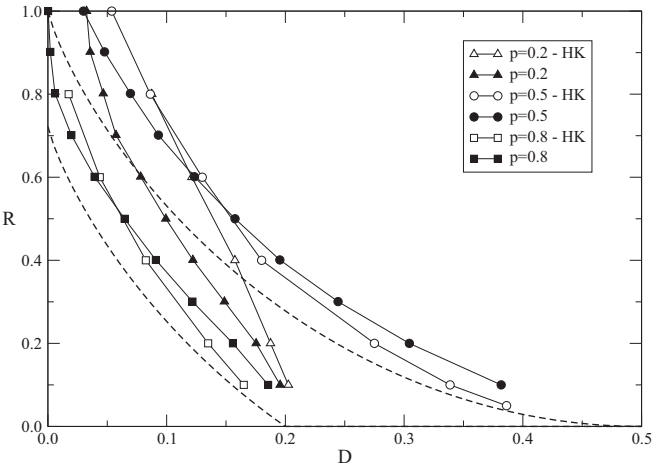

FIG. 2. Performance of the rOnMP-based compression scheme for different levels of bias. Dashed lines in this and the following figures represent the theoretical bounds; the top line for bias $p = 0.5$ and the lower one for $p = 0.2, 0.8$. Solid lines with full symbols are averages over 100 instances using $n = 10^4$ replica, while those with hollow symbols represent results presented in [17] for comparison.

VII. RESULTS

We tested the performance of the rOnMP algorithm against results published in the literature for different bias values of the pattern components. Trials with different weighting options for the averaging of the replicas indicate that, contrary to results obtained for the binary Ising perceptron, there is a considerable difference between results obtained using white and weighted averages. For the suggested compression scheme, choosing the best performing replicon at each step turns out to be much

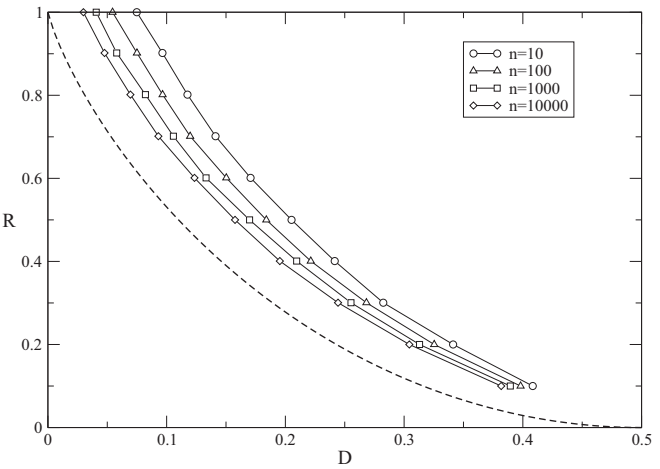

FIG. 3. Performance dependence on the number of replica n for bias $p = 0.5$ and $K = 101$.

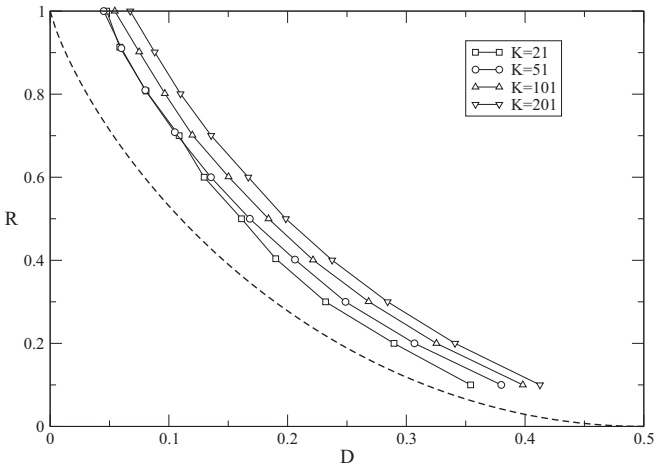

FIG. 4. Performance dependence on the system size K for $p = 0.5$ and $n = 100$.

more efficient than any other choice; it has therefore been used in all the experiments reported below.

Figure 2 shows the performance of the compression scheme in terms of the average bit error or distortion D versus the compression rate R for three different bias values $p = 0.2, 0.5, 0.8$ and system size $K = 101$. The dashed curves show the theoretical bounds, while the solid lines with full symbols show averages over 100 different sets of Gaussian distributed randomly generated patterns with zero mean and unit variance. Each experiment was a run with a total of $n = 10\,000$ replica.

The performance shown in Fig. 2 is better than the results presented in [17] (hollow symbols) for high R values and deteriorates in the lower R regime for $p = 0.5, 0.8$. For $p = 0.2$ our results are better for all R values. As with previous uses of the rOnMP algorithm, performance improves as the number of replica increases. Such an improvement is exemplified in Fig. 3, where we present results for $n = 10, 100, 1000, 10000$ replica for the case of $K = 101$ and $p = 0.5$.

This shows that the main limit for further improvement is computing time. Another notable feature of Fig. 2 is that our results for $p = 0.2$ and $p = 0.8$ are much closer to each other than in previous works. Further experiments indicate that the smaller the size of the system, the closer the curves become using the same algorithm.

Finally, Fig. 4 shows how results change with increasing system size. The graph shows results obtained for $K = 21, 51, 101, 201$ with a bias $p = 0.5$ and $n = 100$ replica. We can see an effect common to most systems with a complex energy landscape. The larger the system, the larger the number of local minima with a higher probability for the algorithm to get trapped; the number of replica needed to attain the same performance increases.

VIII. CONCLUSIONS

We applied the recently introduced replicated online message passing algorithm (rOnMP) [18] to the promising compression method based on a nonlinear perceptron suggested in [15].

The rOnMP algorithm is based on insights from statistical physics and uses a parallel replication of the approximate Bayesian inference procedure known as message passing (MP) to explore the complex energy landscape that characterizes the parameter estimation problem of the nonlinear perceptrons. In addition, the algorithm explores a new way of applying message passing by changing the usual offline MP equations to an online version and by using an expansion for a large system size K.

We showed that our algorithm offers superior performance with respect to conventional MP methods with several additional advantages. (i) The performance of the algorithm is only limited by the available running time as our tests indicate that the larger the number n of replica, the closer to the theoretical performance limits the algorithm gets. (ii) The particular compression scheme we employ suffers from inherent symmetries which prevent the algorithm from converging to the correct value of the binary variables; this usually requires the introduction of a heuristic inertia term [20] in the MP equations. This term is characterized by a constant that has to be fine-tuned. In contrast, our online replicated version of MP does not require any adjustable parameters.

We believe that there is still room for further improvement of the results presented. The natural step is to judiciously choose the path in the example space in a way that maximizes the extraction of information from the set of examples. However, given the complexity of the solution space, this requires new tools and approaches that are currently being investigated.

ACKNOWLEDGMENTS

Support by the Leverhulme trust (F/00 250/M) is acknowledged. We also thank anonymous referees for their useful suggestions.

[1] H. Nishimori, *Statistical Physics of Spin Glasses and Information Processing* (Oxford University Press, Oxford, UK, 2001).
[2] Y. Kabashima and D. Saad, Europhys. Lett. **45**, 97 (1999).
[3] R. H. Swendsen and J.-S. Wang, Phys. Rev. Lett. **57**, 2607 (1986).
[4] E. Marinari and G. Parisi, Europhys. Lett. **19**, 451 (1992).
[5] J. Pearl, *Probabilistic Reasoning in Intelligent Systems* (Morgan Kaufmann, San Francisco, 1988).
[6] M. Mézard and A. Montanari, *Information, Physics, and Computation* (Oxford University Press, Oxford, UK, 2009).
[7] C. E. Shannon, Bell Syst. Tech. J. **27**, 379 (1948).
[8] C. E. Shannon, Bell Syst. Tech. J. **27**, 623 (1948).
[9] T. M. Cover and J. A. Thomas, *Elements of Information Theory* (Wiley-Interscience, New York, 2006).
[10] D. MacKay, *Information Theory, Inference and Learning Algorithms* (Cambridge University Press, Cambridge, UK, 2004).
[11] G. Caire, S. S. Shamai, and S. Verdu, in *ITW 2003, IEEE Information Theory Workshop, Paris, 2003* (IEEE, New York, 2003), pp. 291–295.

[12] S. Ciliberti, M. Mézard, and R. Zecchina, Phys. Rev. Lett. **95**, 038701 (2005).
[13] A. Braunstein, F. Kayhan, and R. Zecchina, in *International Symposium on Information Theory* (IEEE Press, Piscataway, NJ, 2009), pp. 1978–1982.
[14] M. Wainwright, E. Maneva, and E. Martinian, IEEE Trans. Inf. Theory **56**, 1351 (2010).
[15] T. Hosaka, Y. Kabashima, and H. Nishimori, Phys. Rev. E **66**, 066126 (2002).
[16] T. Hosaka and Y. Kabashima, J. Phys. Soc. Jpn. **74**, 488 (2005).
[17] T. Hosaka and Y. Kabashima, Physica A **365**, 113 (2006).
[18] R. C. Alamino, J. P. Neirotti, and D. Saad, Phys. Rev. E **88**, 013313 (2013).
[19] A. Engel and C. van den Broeck, *Statistical Mechanics of Learning* (Cambridge University Press, Cambridge, UK, 2001).
[20] T. Murayama, Phys. Rev. E **69**, 035105 (2004).