Application of Graphical Models to Decoding and Machine Learning

Theodor Misiakiewicz
ICFP, Département de Physique
École Normale Supérieure de Paris
75005 Paris, France
Email: theodor.misiakiewicz@ens.fr

***Abstract*—Graphical Models provide a unifying formalism to deal with two main problems that arise in applications - complexity and uncertainty - by combining graph theory and probabilistic theory. Graphical models are used to represent intricate dependencies among random variables in large-scale statistical systems and the methods developed in this framework have been successfully applied to a wide range of domains, which include machine learning, information theory, bio-informatics, medicine, combinatorial optimization and economics. In this paper, we discuss three applications of the graphical model formalism. First, we present a new general method based on the Bethe approximation and loop calculus to study decoding of low-density parity-check codes. In particular, we show that up to a certain noise threshold, the Bit-Error probability concentrates at zero for the binary erasure and the binary symmetric channel. Then we consider the task of learning from partial observations parameters of a stochastic, dynamic process over a graph. We investigate two algorithms based on the maximum likelihood estimator and the recently introduced dynamical message passing and show that the latter is not only more efficient computationally but it is also robust in the sparse case with respect to the hidden nodes. Finally, we consider the problem of reconstructing parameters of an Ising model from i.i.d. samples. We introduce a new "annealed" convex estimator and argue that it is advantageous over "quenched" pseudo log-likelihood discussed in the literature before. We also briefly describe promising directions for future analysis.**

I. Introduction

A. Motivations

Uncertainty is an unavoidable feature of many real-world applications: the state of a system can be blury/uncertain due to partial or noisy observations, limitation in our understanding of causal relations - e.g. between the observed symptoms and the disease -, or intrinsic non-determinism, e.g. quantum effects. Probability theory provides us with foundational tools for computing likelihoods of different outcomes. It also allows to model random processes, e.g. those emerging in large-scale systems due to complexity of interactions between many identical or similar degrees of freedom. Then, a particularly important question becomes to develop a formalism that compactly stores multivariate distributions in a way which allows efficient computations. Graphical Models (GM) is the technique which helps to achieve this goal.

The key intuition behind GM [1], [2], [3] is modularity: a complex system is often a combination of simpler parts. Each variable depends explicitly only on a small number of other variables, so that the joint probability distribution function is factorized over small subspaces. A graph behind GM provides a graph-based representation of the facorization and the structural dependencies of the distribution. GM allows a compact encoding of high-dimensional distributions. The GM representation may be used for direct tasks, such as computing most probable configuration, marginal distribution or weighted counting (also called partition function), and it is also handy for inverse problem of the Machine Learning type aimed at reconstructing graphs and factor-functions given observations in terms of samples or marginals.

GM and related computational approaches were in the center of the recent developments related to our ever increasing ability to process large data-sets. GM also guided development of modern probabilistic approaches of satistical physics, information theory, machine learning and other related engineering disciplines. Many flavors of GM, such as Hidden Markov models, Ising models, Gaussian models, mixture models, models representing Kalman filters, etc., were introduced and studied. The unifying formalism of GM has allowed generalization of previous tools that have been developed for specific problems. These GM developments have also lead to construction of general-scope algorithms, such as junction tree, linear programming relaxation, max-product, sum-product, expectation propagation etc. Because of its simplicity and versatility, the GM formalism has become a natural first-choice framework to model and design new modular, complex systems.

In the remainder of the Section, we switch gears towards technical descriptions and define GM. We introduce three of the GM most popular classes (directed, undirected and factor graphs) and then briefly summarize the three applications of the GM formalism discussed in the report.

B. Graphical Models

A graph $G = (V, E)$ is formed by a collection of *vertices* $V = (1, \dots, N)$ and a collection of *edges* $E \subset V \times V$. An edge is defined by a pair of vertices $(i, j) \in E$ which can be *directed* (i.e. $(i, j) \neq (j, i)$) or *undirected* (i.e. $(i, j) = (j, i)$). At each vertex $i \in V$ is associated a random variable X_i which takes its value in some space χ_i which can be continuous (e.g. $\chi_i = \mathbb{R}$) or discrete (e.g. $\chi_i = \{0, \dots, m\}$). For any subset $C \subset V$ of the vertices, we will denote $X_C \equiv \{X_i | i \in C\}$ the vector of the random variables in C.

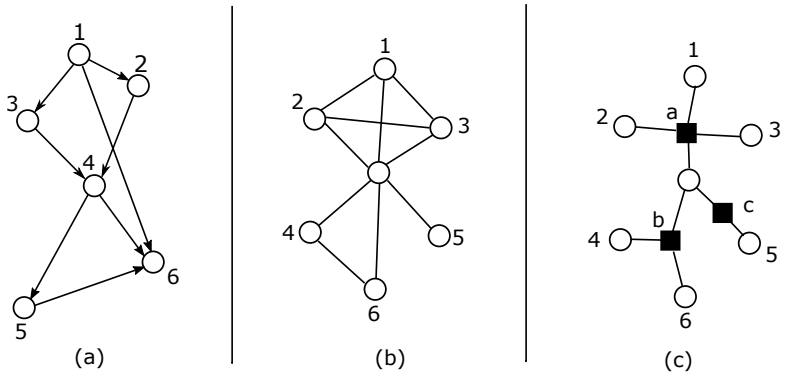

Fig. 1. Three Graphical models classes: (a) directed, (b) undirected and (c) factor graphs

1) Directed graphs: Given a directed graph $G = (V, E)$, for each edge $(i \to j) \equiv (i, j) \in E$ we say that j is the *child* of i and conversely i is a *parent* of j. For each $i \in V$, we denote $\pi(i) \subset V$ the set of all parents of i. A graphical model on this graph is then defined by a collection of positive or null functions $f_i(X_i|X_{\pi(i)})$. The joint probability of the system is given by:

$$P(x_1, \dots x_N) = \frac{1}{Z} \prod_{i \in V} f_i(x_i|x_{\pi(i)}) \tag{1}$$

where Z is the normalization factor, also called *partition function* in the statistical physics community. If the graph is acyclic, the f_i can be choosen normalized (the factor corresponds to the marginal distribution $P(x_i|x_{\pi_i})$ of X_i given $X_{\pi(i)}$): this class of GM is known in statistics as a *bayesian network* (see Figure 1.(a)).

2) Undirected graphs: The graph is now undirected. A *clique* C of the graph is a fully connected subset of V, i.e. $\forall (i, j) \in C \times C, (i, j) \in E$. A graphical model is then defined by a set $\mathcal{C}$ of cliques and positive or null functions associated to these cliques. The joint distribution is given by:

$$P(x_1, \dots x_N) = \frac{1}{Z} \prod_{C \in \mathcal{C}} f_C(x_C). \tag{2}$$

This representation is called *Markov random field* or *Gibbs distribution* and has been widely used in image processing (see Figure 1.(b)).

3) Factor graphs: This representation is an alternative to the undirected graphs which emphasizes the factorization of the distribution. We consider a bipartite graph $G = (V, F, E)$ where $E \subset V \times F$, which can be obtained from an undirected graph by replacing each clique by a node $a \in F$ and the edges (s, a) if and only if the variable X_i participates in the factor a (see Figure 1.(c)). In this new representation, the random variables X_i are associated to the vertices V and the factors f_a to the vertices F, and the factor f_a is a function of the neighboring variables X_i such that $(i, a) \in E$.

C. Applications

1) Decoding: One of the fundamental problems of information theory consists in communicating reliably strings of bits over a noisy channel. During the transmission, each bit may be corrupted by the noise. One can reduce the loss by adding redundancy and using error-correcting codes. When designing the code ensembles (the redundancy), one particular quantity of interest is the bit-error probability: it is the fraction of bits that are on average incorrectly reconstructed. However its computation requires inference on a glassy system which is generically difficult.

In section II, we consider regular low-density parity-check codes over a binary-symmetric channel in the decoding regime. We prove that up to a certain noise threshold the bit-error probability of the bit-sampling decoder converges in mean to zero over the code ensemble and the channel realizations. To arrive at this result we show that the bit-error probability of the sampling decoder is equal to the derivative of a Bethe free entropy. The method that we developed is new and is based on convexity of the free entropy and loop calculus. Convexity is needed to exchange limit and derivatives and the loop series enables us to express the difference between the bit-error probability and the Bethe free entropy. We control the loop series using combinatorial techniques and a first moment method. We stress that our method is versatile and we believe that it can be generalized for LDPC codes with general degree distributions and for asymmetric channels.

2) Learning spreading parameters: Given a directed graph $G = (E, V)$, we consider an activation process on a graph which follows the discrete-time Susceptible-Infected (SI) model [4]. Each vertex can be in either two states: infected (I) or susceptible (S). At every time-step, each infected node can infect one of its susceptible neighbor with a probability α_{ij} associated to the edge $(i \to j) \in E$, while the infected nodes remain infected:

$$\begin{aligned} I(i) + S(j) &\xrightarrow{\alpha_{ji}} I(i) + I(j) \\ I(i) &\to I(i) \end{aligned}$$

An important practical problem consists in reconstructing the diffusion network from data: given a set of cascades - the set of times at which the nodes got infected - on the same graph, can we reconstruct the edges and the spreading parameters α_{ij}? A number of recent papers introduced efficient algorithms for this particular problem, based on the maximization of the likelihood of observed cascades, assuming that the full information for all the nodes in the network is available.

In section III, we focus on a more realistic scenario, in which only a partial information on the cascades is available: either the set of activation times for a limited number of nodes, or the states of nodes for a subset of observation times, or a mixture of both variants. To tackle this problem, we first introduce a framework based on the maximization of the likelihood of the incomplete diffusion trace. However, the evaluation of this incomplete likelihood is a computationally hard problem, and we show that a fast and robust reconstruction of transmission probabilities in sparse networks can be achieved with a new algorithm based on recently introduced dynamic message-passing equations for the spreading processes. The suggested approach can be easily generalized for a large class of discrete and continuous dynamic models, as well as for the cases of dynamically-changing networks and noisy information.

3) Learning Ising Model: An Ising Model on an undirected graph $G = (V, E)$ is defined by the pairwise coulings $J = \{J_{ij}\}_{(i,j)\in E} \in \mathbb{R}^{|E|}$ associated to each edge and magnetic fields $h = \{h_i\}_{i\in V} \in \mathbb{R}^{|V|}$. A binary random variable $\Sigma_i \in \{-1, +1\}$ (spin) is associated to each vertex $i \in V$. The joint distribution of the spins is given by:

$$P_{J,h}(\sigma) = \frac{1}{Z(J)} \exp\left(\sum_{(i,j)\in E} J_{ij}\sigma_i\sigma_j + \sum_{i\in V} h_i\sigma_i\right) \qquad (3)$$

We consider the following problem: given a collection $\Sigma^{(1)}, \ldots, \Sigma^{(M)}$ of independent and identically distributed samples drawn from an Ising Model $\{J, h\}$, reconstruct the underlying graph $G = (V, E)$. Due to its difficulty and its practical importance, the *structure learning problem*, also known as *inverse Ising model problem*, has attracted considerable attention in the past two decades. This year, a work by Guy Bresler [5] has solved a long standing question: can we learn efficiently models without decay of correlation? His greedy algorithm manages to achieve the theoretical sample bound for arbitrary bounded-degree graphs and a runtime of $O(|V|^2)$ without any assumptions on the parameters. However his algorithm, because of the prefactor, remains computationally too expensive to be used in practice.

In section IV, we discuss an alternative approach based on convex estimators which have the advantage to be generic and tractable. We introduce an annealed estimator of "interactions screening" and discuss its potential benefits compared to the quenched pseudo-loglikelihood. We also suggest a new type of regularization based on the spin-flip symmetry.

II. Concentration to Zero Bit-Error Probability for Regular LDPC Codes on the Binary Symmetric Channel: Proof by Loop Calculus

This work has been conducted with M. Vuffray and will be published in the conference proceedings of the 53rd Annual Allerton Conference on Communication, Control, and Computing.

A. Introduction

In 1968 Gallager [6] introduced error-correcting codes based on low-density parity-check (LDPC) matrices. Since then LDPC codes have been proven to be of great practical and theoretical relevance. LDPC codes perform very well under iterative decoding on a broad class of symmetric memoryless channels (BMS) [7], [8] and provably achieve capacity on the binary erasure channel (BEC) [9]. Since 1996 they have been integrated into many industrial standards from wireless communications to computer chips.

An important performance measure of an LDPC code and its associated decoder is the bit-error probability. It is the fraction of bits that are on average incorrectly reconstructed. The bit-error probability of LDPC codes under belief-propagation (BP) decoding is well-understood on BMS channels using the method of density evolution [10]. However it is a more challenging task to control the bit-error probability of the bit maximum a posteriori (MAP) decoder.

Lower and upper bounds on the noise threshold for vanishing bit-MAP error probability have already been derived in Gallager's thesis [6] for a class of BMS channels. These bounds have been improved and generalized for every BMS channels by Shamai and Sason [11].

In this paper we prove that for regular LDPC codes over a BSC channel the "magnetization" or bit-error probability of the bit-sampling decoder vanishes up to a certain threshold. This result also shows that the posterior measure of LDPC codes concentrates over the LDPC ensemble and the noise realizations. To achieve this result we show that the magnetization is asymptotically equal to a perturbed version of the Bethe free entropy. The technique that we present is new and is based on loop calculus or loop series derived by Chertkov and Chernyak [12]. The loop series expresses the difference between a quantity and its Bethe counterpart as a sum over subgraphs. Proving that the loop series vanishes tantamount to control a purely combinatorial object that depends solely on the LDPC graph ensemble. Suboptimal bounds on this object are obtained using McKay's estimates [13] following an idea developed in [14], [15].

The technique that we present has the advantage to be simple and versatile. To emphasize this point we also show that our results can be easily transposed to the BEC. Moreover we stress that our proofs do not rely explicitly on properties of the channel. Hence we believe that this technique can be use to analyze LDPC codes over channels that are not symmetric.

In Section II-B we give a precise definition of the bit-sampling decoder and its associated bit-error probability and we present our main theorems. In Section II-C we derive

the relation between the Bethe free entropy and the bit-error probability and we express the difference using loop calculus. In Section II-C we reduce the loop series to a counting problem that we control with a first moment method. Finally we discuss about future works and possible improvements in Section II-E.

B. Main Results

1) Regular LDPC codes on BMS channels: LDPC codes are defined by a regular bipartite graph $\Gamma = (V, C, E)$ where V is the set of variable nodes, C the set of check nodes and $E = V \times C$ the set of undirected edges. There are $n = |V|$ variable nodes and $m = |C|$ check nodes.

We consider regular LDPC codes with variable-node degrees $l \geq 3$ and check-node degrees $r > l$. The design rate of the code is by definition $R_{\text{des}} = 1 - l/r$.

An LDPC code is generated randomly. The graph Γ is drawn uniformly at random from the ensemble of (l, r) regular bipartite graphs. Throughout the paper we write $\mathbb{E}_{\Gamma}[\cdot]$ the expectation with respect to the ensemble of regular (l, r) bipartite graphs with uniform probability.

Denote the neighbors of a variable node $i \in V$ (resp. a check node $a \in C$) by $\partial i = \{a \in C \mid (i, a) \in E\}$ (resp. by $\partial a = \{i \in V \mid (i, a) \in E\}$). A codeword is a sequence[1] $\underline{\sigma} = \{\sigma_i\}_{i=1}^{n} \in \{-1, 1\}^n$ that satisfies the parity-check sum

$$\prod_{i \in \partial a} \sigma_i = 1, \tag{4}$$

for all check nodes $a \in C$.

We transmit a codeword with uniform prior over a BMS channel with transition probability $q(s_i \mid \sigma_i)$, where the output of the channel could take any real value $s_i \in \mathbb{R}$. The symmetry property of the channel is expressed through the simple relation

$$q(s_i \mid \sigma_i) = q(-s_i \mid -\sigma_i). \tag{5}$$

We assume without loss of generality that the all-zero codeword[2] is transmitted. Hence the output of the channel $\underline{s} = \{s_i\}_{i=1}^{n} \in \mathbb{R}^n$ is i.i.d. with distribution $q(s_i \mid +1)$. The posterior probability that the codeword $\underline{\sigma}$ is sent given that $\underline{s}$ is transmitted reads

$$\mu_{\Gamma}(\underline{\sigma} \mid \underline{s}) = \frac{1}{Z(\Gamma, \underline{s})} \prod_{a \in C} \frac{1}{2} \left(1 + \prod_{i \in \partial a} \sigma_i\right) \prod_{i \in V} q(s_i \mid \sigma_i), \tag{6}$$

where the normalization factor $Z(\Gamma, \underline{s})$ in Equation (6) is the partition function

$$Z(\Gamma, \underline{s}) := \sum_{\underline{\sigma}} \prod_{a \in C} \frac{1}{2} \left(1 + \prod_{i \in \partial a} \sigma_i\right) \prod_{i \in V} q(s_i \mid \sigma_i). \tag{7}$$

2) Concentration of the Bit-Error Probability for the Sampling Decoder: We are interested in the performance of regular LDPC codes with respect to the average bit-error probability of decoding. We consider the bit-sampling decoder

$$\widehat{\sigma}_i^{\text{sampling}}(\underline{s}) := \text{sample } \sigma_i \text{ according to } \sum_{\underline{\sigma} \backslash \sigma_i} \mu(\underline{\sigma} \mid \underline{s}). \tag{8}$$

The bit-error probability of the bit-sampling decoder is directly related to the marginals of the posterior probability (6)

$$P_{\Gamma}^{\text{bit-sampling}} := \frac{1}{2}\left(1 - \mathbb{E}_{\underline{s}}\left[\frac{1}{n}\sum_{i=1}^{n} \langle \sigma_i \rangle_{|\underline{s}}\right]\right), \tag{9}$$

where $\mathbb{E}_{\underline{s}}[\cdot]$ denotes the expectation with respect to the channel output distribution and $\langle \cdot \rangle_{|\underline{s}}$ denotes the average with respect to the posterior probability (6). The expected quantity in Equation (9) is sometimes referred as the averaged magnetization in the physics community.

An important question is to know when the bit-error probability is vanishing in the limit where the codeword length goes to infinity. In this paper we consider two families of symmetric channels, the BEC and the BSC. The BEC has an output alphabet $s_i \in \{-1, 0, 1\}$ and is characterized by transition probabilities

$$q^{\text{BEC}}(1 \mid 1) = 1 - \epsilon,\ q^{\text{BEC}}(0 \mid 1) = \epsilon,\ q^{\text{BEC}}(-1 \mid 1) = 0, \tag{10}$$

where $\epsilon \in [0, 1]$ is the erasure probability. The BSC has binary outputs $s_i \in \{-1, 1\}$ and is characterized by the transition probabilities

$$q^{\text{BSC}}(1 \mid 1) = 1 - p,\ q^{\text{BSC}}(-1 \mid 1) = p, \tag{11}$$

where $p \in [0, 1/2]$ is the flipping probability.

Before we state our theorems we need to introduce the domain

$$\begin{aligned} D(\rho) = \Bigg\{ & \left(x_0, x_c, \underline{y}\right) \in [0,1]^{2+\lfloor r/2 \rfloor} \mid \sum_{t=1}^{\lfloor r/2 \rfloor} y_t \leq 1, \\ & \sum_{t=1}^{\lfloor r/2 \rfloor} \frac{2t}{r} y_t = (1-\rho) x_0 + \rho x_c \Bigg\}. \end{aligned} \tag{12}$$

We also need to introduce the auxiliary function $f : D(\rho) \times [0, 1] \to \mathbb{R}$ defined as follows[3]

$$\begin{aligned} f\left(x_0, x_c, \underline{y}, \rho\right) = & -l h_2\left((1-\rho) x_0 + \rho x_c\right) \\ & + (1-\rho) h_2(x_0) + \rho h_2(x_c) \\ & - \frac{l}{r}\left(1 - \sum_{t=1}^{r} y_t\right) \ln\left(1 - \sum_{t=1}^{r} y_t\right) \\ & - \frac{l}{r} \sum_{t=1}^{r} y_t \ln y_t \\ & + \frac{l}{r} \sum_{t=1}^{\lfloor r/2 \rfloor} y_t \ln \binom{r}{2t}, \end{aligned} \tag{13}$$

[1] As we use concepts from statistical physics it is more convenient to employ the binary alphabet $\{-1, 1\}$ instead of the traditional $\{0, 1\}$.

[2] In the binary alphabet $\{-1, 1\}$, the all-zero codeword is the sequence $\{1, \ldots, 1\}$.

[3] The binary entropy $h_2(p) := -(1-p)\ln(1-p) - p \ln p$ is computed in nat.

and the function $k : [0,1]^4 \to \mathbb{R}$ that reads

$$k\left(x_0, x_c, \rho, p\right) = \left(\rho x_c - (1-\rho)\, x_0\right) \ln\left(\frac{1-p}{p}\right). \qquad (14)$$

Our main contribution are the two theorems stated below which give sufficient conditions on the channel parameters for concentration of the bit-error probability of the sampling decoder.

Theorem 1 (Concentration of the Bit-Error Probability for BEC)**.** *Consider the ensemble of (l, r) regular LDPC codes on a BEC with erasure probability ϵ. If the following function achieves its maximum only at the point*

$$\underset{\left(0, x_c, \underline{y}\right) \in D(\epsilon)}{\operatorname{argmax}} \; f\left(0, x_c, \underline{y}, \epsilon\right) = \{(0,0,0)\},$$

then the bit-error probability of the sampling decoder converges in mean to zero in the large codeword limit

$$\lim_{n\to\infty} \mathbb{E}_{\Gamma,\underline{s}}\left[P_{\Gamma}^{\textit{bit-sampling}}\right] = 0.$$

The same theorem holds for the BSC with a similar condition.

Theorem 2 (Concentration of the Bit-Error Probability for BSC)**.** *Consider the ensemble of (l, r) regular LDPC codes on a BSC with flipping probability p. If the following function achieves its maximum only at the point*

$$\underset{\left(x_0, x_c, \underline{y}\right) \in D(p)}{\operatorname{argmax}} \; f\left(x_0, x_c, \underline{y}, p\right) + k\left(x_0, x_c, p, p\right) = \{(0,0,0)\},$$

then the bit-error probability of the sampling decoder converges in mean to zero in the large codeword limit

$$\lim_{n\to\infty} \mathbb{E}_{\Gamma,\underline{s}}\left[P_{\Gamma}^{\textit{bit-sampling}}\right] = 0.$$

Remark 3. Knowing that $P_{\Gamma}^{\text{bit-sampling}}$ vanishes implies that with high probability the posterior measure (6) concentrates on configurations that are at a Hamming distance $o(n)$ from the all-zero codeword.

We perform the global optimization numerically and we find for a few cases the maximum value of noise ϵ_{loop} and p_{loop} for which Theorem 1 and Theorem 2 hold. The critical values of noise are displayed in Table I for the BEC and in Table II for the BSC.

l	r	R_{des}	ϵ_{BP}	ϵ_{loop}	ϵ_{MAP}	ϵ_{Sh}
3	4	1/4	0.64743	0.7442(9)	0.74601	0.75
3	5	2/5	0.51757	0.5872(4)	0.59098	0.6
3	6	1/2	0.42944	0.4833(6)	0.48815	0.5
4	6	1/3	0.50613	0.5767(2)	0.66565	0.66667

TABLE I
THRESHOLDS FOR SOME REGULAR LDPC CODE ENSEMBLES OVER THE BEC WITH ERASURE PROBABILITY ϵ. THE BELIEF-PROPAGATION THRESHOLD IS ϵ_{BP}, THE MAXIMUM A POSTERIORI THRESHOLD IS ϵ_{MAP}, THE SHANNON THRESHOLD IS ϵ_{SH} AND OUR THRESHOLD IS ϵ_{LOOP}. VALUES OF BP AND MAP THRESHOLDS ARE FROM [16].

l	r	R_{des}	p_{BP}	p_{loop}	p_{MAP}	p_{Sh}
3	4	1/4	0.1669(2)	0.2014(2)	0.2101(1)	0.21450
3	5	2/5	0.1138(2)	0.1146(8)	0.1384(1)	0.14610
3	6	1/2	0.0840(2)	0.0678(9)	0.1010(1)	0.11003
4	6	1/3	0.1169(2)	0.1705(2)	0.1726(1)	0.17395

TABLE II
THRESHOLDS FOR SOME REGULAR LDPC CODE ENSEMBLES OVER THE BSC WITH ERASURE PROBABILITY p. THE BELIEF-PROPAGATION THRESHOLD IS p_{BP}, THE MAXIMUM A POSTERIORI THRESHOLD IS p_{MAP}, THE SHANNON THRESHOLD IS p_{SH} AND OUR THRESHOLD IS p_{LOOP}. VALUES OF BP AND MAP THRESHOLDS ARE FROM [16].

We would expect that for LDPC codes the probability of error vanishes for $\epsilon < \epsilon_{\text{MAP}}$ and $p < p_{\text{MAP}}$. Although the thresholds that we found are reasonably close to ϵ_{MAP} and p_{MAP} for graphs with small degrees, they become worse in the limit of large degrees. A quick inspection of (13) and (14) shows that the functions f/l and k/l become independent of the noise parameter in the limit where l and r go to infinity with a fixed ratio l/r. It implies that p_{loop} and ϵ_{loop} vanish. This behavior is in the opposite direction to what we can expect as in the limit of large degrees $p_{\text{MAP}} \to p_{\text{Sh}}$. In Section II-E we discuss about possible improvements in our analysis in order to make our thresholds tight.

The rest of the paper is organized as follows. In Section II-C we show that the bit-error probability is related to the derivative of the so-called free entropy. Using the loop series, we express the free entropy as a combinatorial sum over subgraphs. In Section II-D we control the loop series with asymptotic estimates on subgraphs and Laplace's method. We prove Theorems 1 and 2 in this section. In Section II-E we discuss future directions and ways to improve and generalize our results.

C. Free Entropy, Bethe Approximation and Loop Series

1) The Free Entropy and its Relation to the Bit-Error Probability: The bit-error probability (9) is related to a "perturbed" version of the partition function (7). Let $\eta \in \mathbb{R}$ be the perturbation parameter entering in the perturbed partition function

$$Z\left(\Gamma, \underline{s}, \eta\right) := \sum_{\underline{\sigma}} \prod_{a\in C} \frac{1}{2}\left(1 + \prod_{i\in\partial a} \sigma_i\right) \prod_{i\in V} q\left(s_i \mid \sigma_i\right) e^{\eta(\sigma_i - 1)}. \qquad (15)$$

Note that $Z\left(\Gamma, \underline{s}, \eta\right)$ is a non-increasing function of η and $Z\left(\Gamma, \underline{s}, 0\right)$ is the original partition function (7).

The free entropy is the (normalized) logarithm of the partition function (15)

$$\phi\left(\Gamma, \underline{s}, \eta\right) := \frac{1}{n} \ln Z\left(\Gamma, \underline{s}, \eta\right). \qquad (16)$$

A direct computation shows that the derivative of the free entropy with respect to its perturbation parameter reads

$$\left.\frac{\partial}{\partial\eta}\phi\left(\Gamma, \underline{s}, \eta\right)\right|_{\eta=0} = \frac{1}{n}\sum_{i=1}^{n} \left\langle \sigma_i \right\rangle_{\underline{s}} - 1. \qquad (17)$$

Therefore the bit-error probability is related to the average entropy through the following relation

$$\frac{\partial}{\partial \eta} \mathbb{E}_{\underline{s}} \left[\phi \left(\Gamma, \underline{s}, \eta \right) \right] \Big|_{\eta=0} = -2 P_{\Gamma}^{\text{bit-sampling}}. \tag{18}$$

Since $Z\left(\Gamma, \underline{s}, \eta\right)$ is a non-increasing function of η, the free entropy is non-increasing as well. Moreover the free entropy is a convex function of η as it can easily be verified by taking twice the derivative with respect to η. It implies that in order to show concentration of the bit-error probability it is sufficient to prove that there exists $\widetilde{\eta} < 0$ independent of n such that $\mathbb{E}_{\Gamma, \underline{s}} \left[\phi \left(\Gamma, \underline{s}, \widetilde{\eta} \right) \right] \to 0$. If this condition is true then, thanks to monotonicity, the limit is also equal to zero for all $\eta \in [\widetilde{\eta}, \infty[$. Finally convexity of the free entropy enables us to exchange limit and derivative (see [17, p. 203]).

In order to prove that the free entropy vanishes we decompose it into two contributions: the Bethe free entropy that can be computed explicitly and the so-called loop series that is a sum over subgraphs of Γ. Using a first moment method and combinatorial tools from graph theory, we show that with high probability the loop series vanishes in the large codeword limit. The last statement implies that the free entropy is equal to the Bethe free entropy.

2) The Bethe Approximation : The Bethe free entropy is an approximation of the free entropy (16). It is defined as a functional over "messages" that are probability distributions $\nu_{i \to a}\left(\sigma_i\right)$, $\widehat{\nu}_{a \to i}\left(\sigma_i\right)$ associated with the directed edges $i \to a$, $a \to i$ of the graph. The messages satisfy the so-called belief-propagation (BP) equations. For the free entropy (16) the BP equations take the following form

$$\begin{aligned} \widehat{\nu}_{a \to i}\left(\sigma_i\right) &\propto \sum_{\underline{\sigma}_{\partial a} \backslash \sigma_i} \frac{1}{2} \left(1 + \prod_{i \in \partial a} \sigma_i \right) \prod_{j \in \partial a \backslash i} \nu_{j \to a}\left(\sigma_i\right) \\ \nu_{i \to a}\left(\sigma_i\right) &\propto e^{\eta(\sigma_i - 1)} q\left(s_i \mid \sigma_i\right) \prod_{b \in \partial i \backslash a} \widehat{\nu}_{b \to i}\left(\sigma_i\right), \end{aligned} \tag{19}$$

where the symbol $\propto$ denotes equality up to a normalization factor.

The Bethe free entropy evaluated at a fixed point of the BP equations is a sum of local contributions from nodes and edges of the graph $\Gamma = (V, C, E)$

$$\phi^{\text{Bethe}}_{(\underline{\nu}, \underline{\widehat{\nu}})}\left(\Gamma, \underline{s}, \eta\right) := \frac{1}{n} \sum_{a \in C} F_a + \frac{1}{n} \sum_{i \in V} F_i - \frac{1}{n} \sum_{(i,a) \in E} F_{ia}, \tag{20}$$

where

$$\begin{aligned} F_a &= \ln \left(\sum_{\underline{\sigma}_{\partial a}} \frac{1}{2} \left(1 + \prod_{i \in \partial a} \sigma_i \right) \prod_{j \in \partial a} \nu_{j \to a}\left(\sigma_i\right) \right) \\ F_i &= \ln \left(\sum_{\sigma_i} e^{\eta(\sigma_i - 1)} q\left(s_i \mid \sigma_i\right) \prod_{b \in \partial i} \widehat{\nu}_{b \to i}\left(\sigma_i\right) \right) \\ F_{ia} &= \ln \left(\sum_{\sigma_i} \nu_{i \to a}\left(\sigma_i\right) \widehat{\nu}_{a \to i}\left(\sigma_i\right) \right). \end{aligned} \tag{21}$$

Note that once a fixed-point of the BP equations (19) is found, computing the Bethe free entropy (20) is a computationally easy task.

3) Corrections to the Bethe Free Entropy: the Loop Series: The difference between the free entropy and the Bethe free entropy can be expressed with the so-called loop series derived by Chertkov and Chernyak [12]. It takes the form of the logarithm of a weighted sum over subgraphs of Γ. These subgraphs are called "loops" for they have no dangling edges. Note that if Γ is a tree no such subgraph exists and we recover the well-known result that the Bethe free entropy is exact on trees. We recall that a subgraph $g = (V_g, C_g, E_g)$ of $\Gamma = (V, C, E)$ is any graph with vertex set $V_g \subset V$, factor node set $C_g \subset C$ and edge set $E_g \subset (V_g \times C_g) \cap E$. For simplicity we denote the relation "g is a subgraph of Γ" with the inclusion symbol $g \subset \Gamma$. We also denote the induced neighborhood in g of a variable node $i \in V_g$ (resp. check node $a \in C_g$) by $\partial_g i = \partial i \cap V_g$ (resp. by $\partial_g a = \partial a \cap C_g$). The set of "loops" consists of any non-empty subgraphs, not necessarily connected, with no degree one variable-node and no degree one check-node

$$\mathcal{L}_{\Gamma} := \left\{ g \subset \Gamma \mid \forall i \in V_g, |\partial_g i| \geq 2 \text{ and } \forall a \in C_g, |\partial_g a| \geq 2 \right\}. \tag{22}$$

The difference between the free entropy and the Bethe free entropy is related to the loop series through the following equation

$$\phi\left(\Gamma, \underline{s}, \eta\right) - \phi^{\text{Bethe}}_{(\underline{\nu}, \underline{\widehat{\nu}})}\left(\Gamma, \underline{s}, \eta\right) = \frac{1}{n} \ln \left(Z^{\text{loop}}_{(\underline{\nu}, \underline{\widehat{\nu}})} \right), \tag{23}$$

where the argument of the logarithm is a weighted sum over loops

$$Z^{\text{loop}}_{(\underline{\nu}, \underline{\widehat{\nu}})} := 1 + \sum_{g \in \mathcal{L}_{\Gamma}} K_{(\underline{\nu}, \underline{\widehat{\nu}})}\left(g\right). \tag{24}$$

The weight function over loops depends on the BP fixed point at which the Bethe free entropy is evaluated and can be expressed as a product over the nodes inside a loop

$$K_{(\underline{\nu}, \underline{\widehat{\nu}})}\left(g\right) := \prod_{i \in V_g} \kappa_i \prod_{a \in C_g} \kappa_a. \tag{25}$$

The factors κ_i and κ_a entering in (25) depend only on messages that are associated with edges neighboring the nodes $i \in V_g$ and $a \in C_g$

$$\begin{aligned} \kappa_i &:= \left(\sum_{\sigma_i} q\left(s_i \mid \sigma_i\right) e^{\eta(\sigma_i - 1)} \prod_{a \in \partial i} \widehat{\nu}_{a \to i}\left(\sigma_i\right) \right)^{-1} \\ &\times \left(\sum_{\sigma_i} q\left(s_i \mid \sigma_i\right) e^{\eta(\sigma_i - 1)} \prod_{a \in \partial i \backslash \partial_g i} \widehat{\nu}_{a \to i}\left(\sigma_i\right) \right. \\ &\times \left. \prod_{a \in \partial_g i} \sigma_i \nu_{i \to a}\left(-\sigma_i\right) \right), \end{aligned} \tag{26}$$

and

$$\kappa_a := \left(\sum_{\underline{\sigma}_{\partial a}} \left(1+\prod_{i\in\partial a}\sigma_i\right)\prod_{i\in\partial a}\nu_{i\to a}(\sigma_i)\right)^{-1} \times \left(\sum_{\underline{\sigma}_{\partial a}} \left(1+\prod_{i\in\partial a}\sigma_i\right)\prod_{i\in\partial a\setminus\partial_g a}\nu_{i\to a}(\sigma_i) \times \prod_{i\in\partial_g a}\sigma_i\widehat{\nu}_{a\to i}(-\sigma_i)\right). \quad (27)$$

For a complete derivation of the loop series for graphical models associated with linear codes, we refer the reader to [18].

4) The Decoding Regime and its BP Fixed-Point: Note that the loop series, as well as the Bethe free entropy, are functions of fixed-points of the BP equations (19). The fixed-point associated with the decoding regime is the ferromagnetic fixed-point

$$\widehat{\nu}^{+}_{a\to i}(\sigma_i) = \quad \nu^{+}_{i\to a}(\sigma_i) = \quad \frac{1+\sigma_i}{2}. \quad (28)$$

One can easily see that ferromagnetic messages (28) satisfy the BP equations (19) regardless of the channel considered and of the value of the perturbation parameter $\eta \in \mathbb{R}$. The ferromagnetic fixed-point (28) describes a state for which the most likely configuration is the all-zero codeword i.e. $\sigma_i = +1$. This is the reason why this fixed-point is associated with the decoding regime.

The Bethe free entropy (20) evaluated at the ferromagnetic fixed-point simply reads

$$\phi^{\text{Bethe}}_{+}(\Gamma,\underline{s},\eta) = \frac{1}{n}\sum_{i\in V}\ln\left(q\left(s_i \mid +1\right)\right). \quad (29)$$

The factors entering in the weight function (25) are computed using Equations (27) for check nodes

$$\kappa_a = \begin{cases} 1 & |\partial_g a| \text{ is even} \\ 0 & |\partial_g a| \text{ is odd} \end{cases}, \quad (30)$$

and Equation (26) for variable nodes

$$\kappa_i = \begin{cases} (-1)^l e^{-2(\lambda(s_i)+\eta)} & |\partial_g i| = l \\ 0 & |\partial_g i| < l \end{cases}, \quad (31)$$

where in the last expression we have used the half log-likelihood variables

$$\lambda(s_i) := \frac{1}{2}\ln\frac{q\left(s_i \mid +1\right)}{q\left(s_i \mid -1\right)}. \quad (32)$$

Based on the expression of the factors (30) and (31), the only subgraphs with a non-zero weight are those with an induced variable-node degree equal to l and even induced check-node degree. This motivates the definition of the ferromagnetic loops ensemble

$$\mathcal{L}^{+}_{\Gamma} = \{g \in \mathcal{L}_{\Gamma} \mid \forall i, a \in g, |\partial_g i| = l \text{ and } |\partial_g a| \text{ is even}\}. \quad (33)$$

A loop that is not an element of the ferromagnetic ensemble has a zero weight. Moreover the weight of a ferromagnetic loop is always non-negative

$$K_{+}(g) = \exp\left(-2\eta|V_g| - 2\sum_{i\in V_g}\lambda(s_i)\right) \geq 0. \quad (34)$$

In order to see that $K_{+}(g)$ is non-negative, notice that a sign is only associated with the factors κ_i and is equal to $(-1)^l$. Therefore a loop can only have a negative weight if the product $l|V_g|$ is odd. Note that this product is the number of edges in a loop counted from the variable-node perspective. Therefore it should be equal to the number of edges counted from the check-node perspective

$$l|V_g| = \sum_{a\in C_g}|\partial_g a|. \quad (35)$$

Since for a ferromagnetic loop $|\partial_g a|$ is always even, $l|V_g|$ is also even and the weight of a loop is always non-negative.

Using Equations (23) and (29) we can express the average free entropy (16) in the simple form

$$\mathbb{E}_{\Gamma,\underline{s}}\left[\phi(\Gamma,\underline{s},\eta)\right] = \mathbb{E}_{\Gamma,\underline{s}}\left[\frac{1}{n}\ln\left(1+\sum_{g\in\mathcal{L}^{+}_{\Gamma}}K_{+}(g)\right)\right] + \int dsq\left(s \mid 1\right)\ln\left(q\left(s \mid 1\right)\right). \quad (36)$$

Note that Equation (36) is valid for all BMS channels regardless of the noise parameter. However we can only expect that the ferromagnetic loop-series vanishes in the decoding regime.

D. First Moment Method on the Loop Series

We use a first moment method to prove that the ferromagnetic loop-series in Equation (36) vanishes. In our case it is based on Jensen's inequality and consists of permuting the expectation over the graph ensemble and the logarithm in Equation (36).

Note that we cannot permute the expectation over the channel output realizations and the logarithm. It is easy to see that over the channel output realizations a loop has an expected weight (34) that increases exponentially fast for $\eta < 0$

$$\mathbb{E}_{\underline{s}}\left[K_{+}(g)\right] = e^{-\eta|V_g|}. \quad (37)$$

This is because the loop series is dominated by events for which most of the bits are corrupted and have negative half log-likelihood (32). These events are rare but give rise to an exponentially large weight.

Therefore we estimate the expectation of the loop series over the ensemble of regular (l,r) bipartite graphs for a fixed output realization of the channel.

1) Probability Estimates on Graphs: For a given channel realization $\underline{s}$ of the BEC (resp. BSC) call V_c the set of variable nodes with $s_i = 0$ (resp. $s_i = -1$) and call V_0 the set of variable nodes $i \in V$ with $s_i = 1$ (resp. $s_i = 1$). The set V_0 contains bits that have been correctly transmitted and V_c contains bits that have been corrupted. We denote the fraction of correctly transmitted bits by $(1-\rho) = |V_0|/n$ and we denote the fraction of corrupted bits by $\rho = |V_c|/n$. We recall that the total number of variable nodes is $n = |V|$ and the total number of check nodes is $m = |C|$.

We decompose the set of ferromagnetic loops (33) into subsets of loops having the same "type". The type of a loop $g \in \mathcal{L}_\Gamma^+$ is the triplet $\left(x_0, x_c, \underline{y}\right) \in [0,1]^{2\times\lfloor r/2\rfloor}$ where $x_0 = |V_0 \cap V_g|/n$ is the fraction of correctly transmitted variable nodes in the loop, $x_c = |V_c \cap V_g|/n$ is the fraction of corrupted variable nodes in the loop and $\underline{y} = \{y_t\}_{t=1}^{\lfloor r/2\rfloor}$ is the fraction of check nodes with degree $2t$. The set of loops of type $\left(x_0, x_c, \underline{y}\right)$ is denoted by $\Omega\left(x_0, x_c, \underline{y}\right)$.

Not all value of $\left(x_0, x_c, \underline{y}\right)$ are admissible loop types. The fraction of check nodes inside a loop is upper bounded by 1. Moreover counting edges from the variable-node perspective or from the check-node perspective obviously gives the same number. Therefore types that are admissible belong to the following set already introduced in Section II-B, Eq. (12)

$$\begin{aligned} D(\rho) \quad = \quad & \left\{ \left(x_0, x_c, \underline{y}\right) \in [0,1]^{2+\lfloor r/2\rfloor} \mid \sum_{t=1}^{\lfloor r/2\rfloor} y_t \leq 1, \right. \\ & \left. \sum_{t=1}^{\lfloor r/2\rfloor} \frac{2t}{r} y_t = (1-\rho)\, x_0 + \rho x_c \right\}. \end{aligned} \tag{38}$$

The weight (34) of a loop $g \in \Omega\left(x_0, x_c, \underline{y}\right)$ is only a function of its type $K_+(g) \equiv K_+(x_0, x_c)$. Using the specific expression of the half log-likelihood (32) for each channels we find the explicit form of the weight function for the BEC

$$K_+^{\text{BEC}}(x_0, x_c) = \begin{cases} \exp\left(-2n\eta x_c \rho\right) & x_0 = 0 \\ 0 & x_0 > 0 \end{cases}, \tag{39}$$

and for the BSC

$$\begin{aligned} K_+^{\text{BSC}}(x_0, x_c) \quad = \quad & \exp\left(-2n\eta\left(x_0(1-\rho) + x_c\rho\right)\right. \\ & \left. +nk\left(x_0, x_c, \rho, p\right)\right), \end{aligned} \tag{40}$$

where $k(x_0, x_c, \rho, p)$ is the auxiliary function introduced in Section II-B, Eq. (14)

$$k(x_0, x_c, \rho, p) = \left(\rho x_c - (1-\rho)\, x_0\right) \ln\left(\frac{1-p}{p}\right). \tag{41}$$

Therefore the expected value of the loop series over the graph ensemble can be expressed only through loop types

$$\begin{aligned} \mathbb{E}_\Gamma\left[\sum_{g\in\mathcal{L}_\Gamma^+} K_+(g)\right] \quad = \quad & \sum_{\left(x_0,x_c,\underline{y}\right)\in D(\rho)} K_+(x_0, x_c) \\ & \times \quad \mathbb{E}_\Gamma\left[\left|\Omega\left(x_0, x_c, \underline{y}\right)\right|\right]. \end{aligned} \tag{42}$$

The expected number of loops with prescribed type $\left(x_0, x_c, \underline{y}\right)$ is upper bounded using McKay's combinatorial estimate[4] [13] for subgraphs with specified degrees

$$\begin{aligned} \mathbb{E}_\Gamma\left[\left|\Omega\left(x_0, x_c, \underline{y}\right)\right|\right] \quad \leq \quad & n^{\delta_{l,r}} \begin{pmatrix} nl \\ nl\left(x_0(1-\rho) + x_c\rho\right) \end{pmatrix}^{-1} \\ & \times \quad \begin{pmatrix} n(1-\rho) \\ nx_0(1-\rho) \end{pmatrix} \begin{pmatrix} n\rho \\ nx_c\rho \end{pmatrix} \\ & \times \quad \begin{pmatrix} m \\ my_1, \ldots, my_{\lfloor r/2\rfloor} \end{pmatrix} \\ & \times \quad \prod_{t=1}^{\lfloor r/2\rfloor} \begin{pmatrix} r \\ 2t \end{pmatrix}^{my_t}, \end{aligned} \tag{43}$$

where $\delta_{l,r}$ is a constant that depends only on l and r. McKay's estimate has the advantage to have an asymptotically tight growth rate when n goes to infinity.

It remains to prove that the average loop series (42) with the bound (43) vanishes in the large n limit.

2) Laplace's Method and Proof of Theorems: The loop series (42) is dominated by loop types that contribute to the sum with the biggest exponential growth. We apply Laplace's method in order to characterize the biggest exponent.

Using Stirling inequalities

$$e^{\frac{1}{12n+1}} \leq \frac{n!}{\sqrt{2\pi n} e^{-n} n^n} \leq e^{\frac{1}{12n}}, \tag{44}$$

we find an asymptotically tight upper bound on the estimate (43)

$$\mathbb{E}_\Gamma\left[\left|\Omega\left(x_0, x_c, \underline{y}\right)\right|\right] \leq C_{l,r} n^{\delta'_{l,r}} \exp\left(nf\left(x_0, x_c, \underline{y}, \rho\right)\right), \tag{45}$$

where $C_{l,r}$ and $\delta'_{l,r}$ are just numerical constants and $f\left(x_0, x_c, \underline{y}, \rho\right)$ is the auxiliary function introduced in Section II-B, Eq. (4)

$$\begin{aligned} f\left(x_0, x_c, \underline{y}, \rho\right) \quad = \quad & -lh_2\left(\left(1-\rho\right)x_0 + \rho x_c\right) \\ & + \quad \left(1-\rho\right) h_2\left(x_0\right) + \rho h_2\left(x_c\right) \\ & - \quad \frac{l}{r}\left(1 - \sum_{t=1}^{r} y_t\right) \ln\left(1 - \sum_{t=1}^{r} y_t\right) \\ & - \quad \frac{l}{r} \sum_{t=1}^{r} y_t \ln y_t \\ & + \quad \frac{l}{r} \sum_{t=1}^{\lfloor r/2\rfloor} y_t \ln \begin{pmatrix} r \\ 2t \end{pmatrix}. \end{aligned} \tag{46}$$

Combining Equations (39), (40) and (45), we show that the leading exponent in Equation (42) is for the BEC

$$\alpha^{\text{BEC}}(\rho, \eta) = \max_{\left(0,x_c,\underline{y}\right)\in D(\rho)} f\left(0, x_c, \underline{y}, \rho\right) - 2\eta x_c \rho, \tag{47}$$

[4]McKay's bound in its original form is only applicable for subgraphs of size less than $n - 4r^2$. We refer to [14] for a careful analysis.

and for the BSC

$$\alpha^{\text{BSC}}(\rho,\eta) = \max_{(x_0,x_c,\underline{y})\in D(\rho)} \left(-2\eta\left(x_0(1-\rho)+x_c\rho\right) + f\left(x_0,x_c,\underline{y},\rho\right)+k\left(x_0,x_c,\rho,p\right)\right). \quad (48)$$

Notice that for all ρ and η the exponent $\alpha^{\text{BEC/BSC}}(\rho,\eta)$ is non-negative. This is easily verified by evaluating the objective function at $\left(x_0,x_c,\underline{y}\right)=(0,0,0)$. Therefore the bit-error probability vanishes if $\alpha^{\text{BEC/BSC}}(\rho,\eta)$ is equal to zero for all η in a neighborhood of zero. The next Lemma shows that in fact only the maximization at $\eta=0$ is important.

Lemma 4. *If the maximum of* (47) *(resp.* (48)*) is uniquely achieved in* $\left(x_0,x_c,\underline{y}\right)=(0,0,0)$ *for* $\eta=0$*, then there exists* $\widetilde{\eta}<0$ *such that* $\alpha^{BEC}(\rho,\eta)=0$ *(resp.* $\alpha^{BSC}(\rho,\eta)=0$*) for all* $\eta\in]\widetilde{\eta},\infty[$.

Proof: See Appendix A-A ∎

In order to prove Theorems 1 and 2, we need to show that small variations around ρ do not change $\alpha^{\text{BEC}}(\rho,0)$ and $\alpha^{\text{BSC}}(\rho,0)$. This is guaranteed by the following Lemma.

Lemma 5. *For all* $\rho\in[0,1]$*, if* $\alpha^{BEC}(\rho,0)=0$ *(resp.* $\alpha^{BSC}(\rho,0)=0$*) and the maximum of* (47) *(resp.* (48)*) is uniquely achieved at* $\left(x_0,x_c,\underline{y}\right)=(0,0,0)$*, there exists* N *sufficiently large such that*

$$\forall n\geq N,\ \forall\delta\in\left[-\sqrt{\frac{\ln n}{n}},\sqrt{\frac{\ln n}{n}}\right],\ \alpha^{BEC/BSC}(\rho+\delta,0)=0$$

Proof: See Appendix A-B ∎

We are now in position to prove our main theorems.

Proof of Theorem 1:

Let ϵ be the probability of error of the BEC. First notice that the perturbed partition function (15) is trivially lower bounded by 1 and upper bounded by $2^n e^{2n|\eta|}$. This implies that the free entropy (16) remains finite

$$0\leq\phi(\Gamma,\underline{s},\eta)\leq\ln 2+2|\eta|. \quad (49)$$

Therefore using Equation (36) and the fact that $K_+(g)\geq 0$ we see that the loop series remains finite as well

$$2(\ln 2+|\eta|) \geq \left|\mathbb{E}_{\underline{s}}\left[\phi(\Gamma,\underline{s},\eta)\right]-\int dsq(s\mid 1)\ln\left(q(s\mid 1)\right)\right| = \mathbb{E}_{\underline{s}}\left[\frac{1}{n}\ln\left(1+\sum_{g\in\mathcal{L}_\Gamma^+}K_+(g)\right)\right]. \quad (50)$$

Let A be the following probabilistic event on the channel output realizations

$$A:=\left\{\underline{s}\in\{-1,0,1\}^n\mid\left|\frac{1}{n}\sum_{i=1}^n s_i-(1-\epsilon)\right|\leq\sqrt{\frac{\ln n}{n}}\right\}. \quad (51)$$

Output realizations in A are close to the average output realization.

Using Hoeffding's inequality, we see that the probability of the complementary event A^c vanishes

$$\mathbb{P}_{\underline{s}}[A^c]\leq\frac{2}{n^{-2}}. \quad (52)$$

Combining Jensen's inequality and the trivial bound (50) on the loop series we have the following estimate

$$\mathbb{E}_{\Gamma,\underline{s}}\left[\frac{1}{n}\ln\left(1+\sum_{g\in\mathcal{L}_\Gamma^+}K_+(g)\right)\right]\leq\frac{4}{n^{-2}}(\ln 2+|\eta|) +\mathbb{E}_{\underline{s}}\left[\frac{1}{n}\ln\left(1+\mathbb{E}_\Gamma\left[\sum_{g\in\mathcal{L}_\Gamma^+}K_+(g)\right]\right)\mid A\right]. \quad (53)$$

Since we have conditioned over channel output realizations that are in A, the fraction of corrupted bit is $|\rho-\epsilon|\leq\sqrt{\ln n/n}$. Therefore combining Equation (45), Lemma 4 and Lemma 5 we have that if $\alpha^{\text{BEC}}(\epsilon,0)=0$ is uniquely achieved in $\left(x_0,x_c,\underline{y}\right)=(0,0,0)$ then for all $\eta\in]\widetilde{\eta},\infty[$ and n sufficiently large,

$$\mathbb{E}_{\underline{s}}\left[\frac{1}{n}\ln\left(1+\mathbb{E}_\Gamma\left[\sum_{g\in\mathcal{L}_\Gamma^+}K_+(g)\right]\right)\mid A\right]\leq\frac{1}{n}\ln\left(1+c_3n^{c_4}\right), \quad (54)$$

where c_3 and c_4 are numerical constants independent of n.

We have proved that for all $\eta\in]\widetilde{\eta},\infty[$ with $\widetilde{\eta}<0$ the average free entropy converges in expectation over the regular (l,r) LDPC ensemble

$$\lim_{n\to\infty}\mathbb{E}_\Gamma\left[\left|\mathbb{E}_{\underline{s}}\left[\phi(\Gamma,\underline{s},\eta)\right]-\int dsq(s\mid 1)\ln\left(q(s\mid 1)\right)\right|\right]=0. \quad (55)$$

In particular it implies that the average free entropy over the LDPC ensemble converges

$$\lim_{n\to\infty}\mathbb{E}_{\Gamma,\underline{s}}\left[\phi(\Gamma,\underline{s},\eta)\right]=\int dsq(s\mid 1)\ln\left(q(s\mid 1)\right). \quad (56)$$

Since $\mathbb{E}_{\Gamma,\underline{s}}\left[\phi(\Gamma,\underline{s},\eta)\right]$ is a convex function of η and converges pointwise in a neighborhood of zero, we can exchange the limit and the derivative

$$\begin{aligned}0 &= \frac{\partial}{\partial\eta}\lim_{n\to\infty}\mathbb{E}_{\Gamma,\underline{s}}\left[\phi(\Gamma,\underline{s},\eta)\right]\Big|_{\eta=0}\\ &= \lim_{n\to\infty}\frac{\partial}{\partial\eta}\mathbb{E}_{\Gamma,\underline{s}}\left[\phi(\Gamma,\underline{s},\eta)\right]\Big|_{\eta=0}\\ &= \lim_{n\to\infty}\mathbb{E}_\Gamma\left[\frac{\partial}{\partial\eta}\mathbb{E}_{\underline{s}}\left[\phi(\Gamma,\underline{s},\eta)\right]\right]\Big|_{\eta=0}\\ &= -2\lim_{n\to\infty}\mathbb{E}_\Gamma\left[P_\Gamma^{\text{bit-sampling}}\right], \end{aligned} \quad (57)$$

where in the last line we use Equation (18) that relates the free entropy to the bit-error probability. ∎

Theorem 2 has a proof almost identical to that of Theorem 1.

E. Path Forward

We would like to stress that the techniques developed in this paper are quite general. In particular they do not rely on a special form of channels or on the regular-degree distribution of the LDPC ensemble. Therefore we plan to improve our results in the following ways.

1) Generalization to Arbitrary Degree Distributions: The entire analysis can easily be extended to general degree distributions with bounded degrees. It will simply transform the function (46) that counts subgraphs into a more convoluted object. However extending our results to distributions with unbounded degrees, like for instance Poisson distributions, may be more complicated. One would have to derive an estimate for counting subgraphs in this particular case.

2) Asymmetric Channels: The loop series and the Bethe free entropy for general channels are almost exactly similar than for symmetric channels. For general channels we can no longer assume that the all-zero codeword is transmitted. Instead we have to average the bit-error probability over all possible input codewords $\underline{\tau}$. In this case the weight of a loop remains similar than for symmetric channels. The weight is also non-negative and depends on the generalized half log-likelihood ratio

$$\lambda\left(s_i \mid \tau_i\right) = \frac{1}{2}\log\frac{q\left(s_i \mid \tau_i\right)}{q\left(s_i \mid -\tau_i\right)}, \tag{58}$$

where $\underline{s}$ denotes as usual the channel observations. In order to control the loop series, we will need to perform a conditioned expectation in (53) over joint typical sequences of input codewords and noise realizations.

3) Tight Thresholds: As described in Section II-B, the thresholds that we obtain are not tight. In fact at fixed rate they become worse and converge to zero as the degrees of the graph become large. The reason why we obtain such loose bounds for large degrees comes from the function $f\left(x_0, x_c, \underline{y}, \rho\right)$ defined in (46). This function counts the growing rate of the average number of subgraphs with a prescribed type $\left(x_0, x_c, \underline{y}\right)$

$$f = \lim_{n\to\infty}\frac{1}{n}\ln\left(\mathbb{E}_\Gamma\left[\left|\Omega\left(x_0, x_c, \underline{y}\right)\right|\right]\right). \tag{59}$$

One can verify that if instead of f we use the function

$$\widetilde{f} = \lim_{n\to\infty}\frac{1}{n}\mathbb{E}_\Gamma\left[\ln\left(\left|\Omega\left(x_0, x_c, \underline{y}\right)\right|\right)\right], \tag{60}$$

we obtain tight lower and upper bound on the threshold for vanishing bit-error probability.

The function $\widetilde{f}$ only depends on the random graph ensembles that we consider and does not depend on a particular channel. Computing this function would provide a proof of the exact location of the MAP threshold for an extensive class of channels. However this computation could prove to be a very difficult task.

A way around the problem of computing (60) is to condition the expectation (59) on some rare events with respect to the random graph measure. Note that by Jensen's inequality $\widetilde{f}$ is always upper-bounded by f. This is because the expectation (59) is dominated by rare events that are associated with a large weight $\left|\Omega\left(x_0, x_c, \underline{y}\right)\right|$. Conditioning on these rare events will lead to better estimates of (60) and will provide tighter bounds at least in the limit of large degrees.

III. Efficient reconstruction of transmission probabilities in a spreading process from partial observations

This work has been conducted with A. Lokhov and will be presented at the Conference on Complex Systems 2015 (CSS'15).

A. Introduction

Learning an underlying graphical model from observed data is a long-standing and important practical problem in statistical physics, machine learning and computer science. Recent years have seen a renewed interest in development of fast and efficient algorithms to carry out this reconstruction problem in diverse contexts, such as gene regulatory networks [19], biopolymers' structure determination [20], neuroscience [21] and sociology [22], using the analysis of large datasets which have become available in these fields. An ongoing effort of scientific community has allowed to develop a number of techniques for solving the inverse problem for simple, bud widely applicable models, such as the Ising model in a static [23], [24] and dynamic [25], [26] settings. However, the inference of the parameters in a large class of other models of diffusion type has been less studied so far. Among a broad range of disordered and out-of-equilibrium dynamic models, a particular attention is devoted to cascading processes which are used for modelling phenomena in a large number of domains: epidemic and rumor spreading [27], [28], spreading of information and innovations in real-world and virtual social networks [29], [30], avalanches in magnetic and glassy systems [31], activation cascades in neural networks [32], *etc*.

Contrary to the case of recurrent models, in which the network reconstruction can be achieved with observing one realization of dynamics of sufficient duration [33], [34], learning in the case of cascades with unidirectional (also called progressive) dynamics requires a certain number of independent avalanches with different initial conditions. Given a subset of activation times for several realizations of the spreading process, the reconstruction problem aims to infer the spreading parameters of the model. In the Bayesian framework, a common inference method relies on the maximization of likelihood of observed information. In the case of fully observed cascades, this approach has been indeed suggested in a number of recent papers [35], [36], [37], leading to distributed convex optimization algorithms and outperforming previously suggested ones. However, in the majority of realistic applications, it is very difficult or even practically impossible to monitor the state of each and every node over the whole duration of the diffusion process; hence a need to develop reconstruction algorithms which would be able to infer the parameters of the model in the presence of hidden nodes or incomplete time information on the cascades, as well as being robust with respect to the noise in the observations. Despite the

importance of this problem, the case of incomplete information has been very poorly addressed so far. In the context of kinetic Ising model, the corresponding learning problem has been studied in [38], where a trace over the configurations of the hidden nodes in the likelihood function is performed using the saddle-point approximation in the path-integral approach and the mean-field methods. In the studies of cascading processes, the work [39] addressed the network learning problem in the case where the possibly noisy observations are recorded at a frequency lower than the one inherent to the dynamic process by using relaxation optimization techniques. Finally, the presence of hidden nodes has been considered in [40] for a different problem of identification of the diffusion source, where the computation of the incomplete likelihood leads to a difficult high-dimensional integration problem.

In this Letter, we develop a systematic framework for parameters estimation in a spreading model from incomplete observations. As a first natural step in solving this problem, we introduce two algorithms based on the maximum likelihood estimator (MLE) of incomplete information. However, both schemes require an exponentially large number of operations for an exact solution, which represents an important limitation of the algorithms and makes their use impossible in the cases where a fast online learning is desired. In practice, we approximate the objective function in the second scheme by using a Monte-Carlo sampling which speeds up the algorithm but still requires to process all the data at each step. As an alternative which would allow to considerably improve the computation time, we develop a new algorithm based on recently introduced dynamic message-passing (DMP) equations for the spreading processes [41]. These equations allow for an asymptotically exact computation of marginal probabilities of nodes' activation on loopy-but-sparse networks, and can be used as an approximate tool for computationally hard problems: recently, DMP equations have been applied to the problem of inference of epidemic origin from a given (possible incomplete) snapshot of the epidemic at a certain time [42].

B. Learning parameters from partial observations with maximum likelihood

1) Formulation of the problem: Let $G \equiv (V, E)$ be a connected undirected graph containing N nodes defined by the set of vertices V and the set of edges E. We observe M realizations of cascades c, where each sample Σ^c represents a set of activation times for the nodes in the network $\{\tau_i^c\}_{i\in V}$. However, some information on the cascades can be missing: the full information can be written as $\Sigma = \Sigma_{\mathcal{O}} \cup \Sigma_{\mathcal{H}}$, where $\Sigma_{\mathcal{O}}$ is the observed part of the cascades, and $\Sigma_{\mathcal{H}}$ represents the hidden part. For the sake of simplicity and definiteness, we assume that the activation process follows a discrete-time susceptible-infected (SI) model, which is defined as follows [28]: each node i at time $t \in [0, T]$ can be in one of two states $q_i(t)$: susceptible, $q_i(t) = S$, or infected, $q_i(t) = I$. At each time step, an infected node j can transmit the information to one of its susceptible neighbors i on the interaction graph G with probability α_{ji}, meaning that i changes its state with a probability $P_t(S(i) \to I(i)) = 1 - \prod_{k\in\partial i}(1 - \alpha_{ki}\mathbb{1}[q_k(t) = I])$, where ∂i denotes the set of neighbors of i; once the node is activated, it stays in the infected state forever. Note that the reconstruction problem can be straightforwardly generalized to models with more complicated transition rules, such as SIR model, threshold and rumor spreading models, and even models with recurrent dynamics. The cascades are simulated with the following initial condition: each node is independently drawn as infected with probability $1/N$, meaning that on average there is one "patient zero" at initial time; note, however, that it also means that some cascades have several initial sources, while other cascades are trivial and do not contain any infected nodes.

Our goal is to reconstruct the values of the couplings $\{\alpha_{ij}\}_{(ij)\in E} \equiv G_\alpha$. In the absence of any prior on the underlying model, the Bayes theorem states that

$$G_\alpha = \arg\max P(G_\alpha \mid \Sigma_{\mathcal{O}}) \propto \arg\max P(\Sigma_{\mathcal{O}} \mid G_\alpha), \quad (61)$$

where $\Sigma_{\mathcal{O}} \equiv \{\Sigma_{\mathcal{O}}^c\}_{c\in[1,M]}$. Hence, the task is to estimate efficiently the likelihood function $P(\Sigma_{\mathcal{O}} \mid G_\alpha)$. Note that the formulation (61) is valid for the case where the structure of the graph is unknown (since we can view a network as a fully-connected graph with some couplings equal to zero). However, in what follows and unless stated otherwise, we assume that the network G is known; treating the case of unknown graph with missing information would require some additional assumptions and constraints on the network structure. For the tests involving incomplete observations, we focus for definiteness and without loss of generality on the presence of nodes with hidden information, providing the study of other cases in the Supplementary Information [43] (see the noisy case in Appendix B-C).

2) Maximum likelihood estimator: If the information on all the nodes is available ($\Sigma = \Sigma_{\mathcal{O}}$), an efficient strategy would be to use a consistent maximum likelihood estimator, suggested in [36]. In the discrete formulation, the likelihood of the cascades, $P(\Sigma \mid G_\alpha)$, is given by:

$$P(\Sigma \mid G_\alpha) = \prod_{i\in V} \prod_{1\le c\le M} P_i(\tau_i^c \mid \Sigma_c \backslash \tau_i^c, G_\alpha), \quad (62)$$

where

$$P_i(\tau_i^c \mid \Sigma\backslash\tau_i^c, G_\alpha) = \left(\prod_{t'=0}^{\tau_i^c-2} \prod_{k\in\partial i} (1 - \alpha_{ki}\mathbb{1}[\tau_k^c \le t'])\right) \times \left[1 - \left(\prod_{k\in\partial i} (1 - \alpha_{ki}\mathbb{1}[\tau_k^c \le \tau_i^c - 1])\right)\mathbb{1}[\tau_i^c < T]\right]. \quad (63)$$

The estimation of the transmission probabilities $\widehat{G}_\alpha$ is given by the solution of the following optimization problem:

$$\widehat{G}_\alpha = \arg\min\left(-\log P(\Sigma \mid G_\alpha)\right), \quad (64)$$

which is convex, and can be solved locally for each node i and its neighborhood due to the factorization of the likelihood under assumption of asymmetry of the couplings. In the case of partial observations, we need to consider the reduced MLE,

performing a trace over the hidden nodes:

$$P(\Sigma_{\mathcal{O}} \mid G_\alpha) = \sum_{\{\tau_i\}_{i\in\mathcal{H}}} P(\Sigma \mid G_\alpha). \quad (65)$$

An exact evaluation of (65) is a computationally difficult high-dimensional problem with complexity proportional to T^H, where H is the number of hidden nodes. The integration can be generically approximated by sampling the phase space: an efficient importance sampling was suggested in [40], to evaluate (65). However, in our case, the total likelihood is dominated by the cascades with a small likelihood, which are difficult to sample. Furthermore, the problem is very unstable with respect to the sampling error, and in practice one need an exact integration, which leads to an algorithm with complexity $O(NMT^H)$ at each step of optimization procedure; see [43] for more details.

3) Heuristic two-stage algorithm: In order to avoid the exponential complexity in H and to keep the nice convexity properties of the full MLE, we introduce a modification of the scheme above which we also use as a benchmark (this algorithm will be referred to as HTS algorithm). The idea is to use two alternating stages at each step of optimization. First, we complete the missing information in the cascades $\Sigma_{\mathcal{H}}$ using the current estimation of the couplings $\widehat{G}_\alpha$, i.e. update the activation times of the hidden variables as follows:

$$\widehat{\Sigma}_{\mathcal{H}} = \arg\max P(\Sigma \mid \widehat{G}_\alpha). \quad (66)$$

We approximate the inference problem (66) with a Monte-Carlo importance sampling: the hidden times are sampled by simulating cascades with the current guess of variables and starting from the known sources. Second, we can solve a convex optimization problem (64) using the full $\Sigma = \Sigma_{\mathcal{O}} \cup \widehat{\Sigma}_{\mathcal{H}}$, thus obtaining a new estimation of $\widehat{G}_\alpha$; the procedure is repeated until convergence. In order to solve exactly the first step, one need an exponential in H number of samples. However we see that a much smaller number of samples is needed in practice and the scheme converges in a small number of steps. The complexity of one iteration step of this algorithm is $O(NML_{H,T})$ [43].

C. Dynamic message-passing algorithm:

A way to quantify the interdependence of activation times of different nodes is to use the dynamic equations that contain information about the correlations occurring in the spreading process. The suggested algorithm is based on the dynamic message-passing equations for diverse dynamic processes [41]. According to the DMP equations for the SI-type activation process, the marginal probability $m^i(\tau_i)$ of activation of node $i \in V$ at time τ_i can be computed as

$$m^i(0) = 1 - P_S^i(0), \quad (67)$$

$$m^i(\tau_i) = P_S^i(0)\left[\prod_{k\in\partial i} \theta^{k\to i}(\tau_i - 1) - \prod_{k\in\partial i} \theta^{k\to i}(\tau_i)\right] \quad (68)$$

for $\tau_i > 0$, where $P_S^i(0)$ is the probability that node i is initialized in the state S. The quantities $\theta^{k\to i}(t)$ are computed iteratively using the following expressions:

$$\theta^{k\to i}(t) = \theta^{k\to i}(t-1) - \alpha_{ki}\phi^{k\to i}(t-1), \quad (69)$$

$$\phi^{k\to i}(t) = (1-\alpha_{ki})\phi^{k\to i}(t-1) + P_S^k(0)\prod_{l\in\partial k\setminus i} \theta^{l\to k}(t-1) - P_S^k(0)\prod_{l\in\partial k\setminus i} \theta^{l\to k}(t), \quad (70)$$

with the initial conditions $\theta^{i\to j}(0) = 1$ and $\phi^{i\to j}(0) = 1 - P_S^i(0)$. The proof that these equations are exact on trees and empirical studies of performance on sparse real-world networks are discussed in [41].

Let us now explain the reconstruction algorithm based on the DMP equations. Given the data on the cascades $\Sigma_{\mathcal{O}}$, we can compute the empirical initial conditions $P_S^i(0)$ and marginal probabilities $m_*^i(\tau_i)$, simply given by the averages of activation times over different cascades at all nodes for which the information is known. The idea, reminiscent of what has been previously used in online learning of parameters in the context of artificial neural networks [44], is to adjust the transmission probabilities G_α in order to minimize the mismatch J between the DMP-estimated and available empirical marginals at each time step:

$$J = \sum_{t=0}^{T-1} J(t) = \sum_{t=0}^{T-1}\sum_{i\in\mathcal{O}} \frac{1}{2}[m_*^i(t) - m^i(t)]^2. \quad (71)$$

To this end, we use a simple gradient descent: starting from some initial distribution of transmission probabilities, the couplings are updated as $\alpha_{rs}^{(t+1)} \leftarrow \alpha_{rs}^{(t)} - \epsilon\frac{\partial J(t)}{\partial \alpha_{rs}}$, where ϵ is the learning rate. The derivatives of the cost function (71) with respect to couplings can be expressed through $\frac{\partial \theta^{k\to i}(t)}{\partial \alpha_{rs}} \equiv p_{rs}^{k\to i}(t)$ and $\frac{\partial \phi^{k\to i}(t)}{\partial \alpha_{rs}} \equiv q_{rs}^{k\to i}(t)$, for which the DMP-like equations can be written using an explicit derivation of the equations (69)-(70) [43] (see Appendix B-A). The update of the transmission probabilities is restarted from time zero until the convergence of the algorithm. Because of the averaging over the cascades, the resulting computational complexity of an iteration step of the DMP algorithm is independent on M and is $O(NdT)$, where d is the average degree of the graph.

D. Performance of reconstruction algorithms

Although the algorithms described above are designed for the case of incomplete information, for the validation purposes we first test their performance in the case of fully observed cascades. In the Fig. 2, we present results for the mean error of reconstruction per coupling on a tree graph and on a connected component of an artificially-generated random graph with a Pareto power-law degree distribution with a shape parameter 2.5 and minimum value parameter 1. The algorithms were initialized at $\alpha_{ij} = 0.5$ for all edges (i, j). On the tree and power-law network, we see that the MLE demonstrates better performance. The DMP algorithm, in the tree case, converges slower towards the right couplings, as expected because DMP equations are exact on a tree. As expected, the DMP algorithm provides a poorer reconstruction on a small random graph which contains loops of small length, cf.

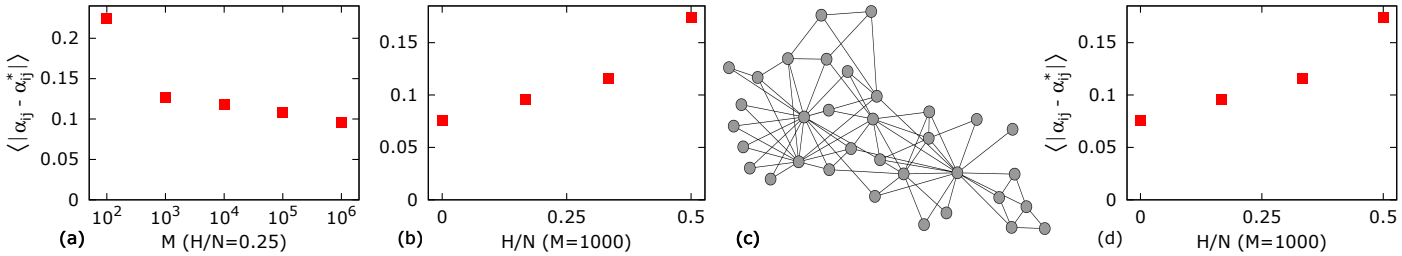

Fig. 3. (Color online) [The figure is not yet complete] Comparison of mean error on the reconstructed couplings as a function of the number of fully observed cascades M for a tree network with $N = 50$ (left figure) and a connected component of a power-law network with $N = 53$ (right figure). Red squares are data points for the DMP algorithm, blue circles correspond to the reconstruction with ML algorithm. Scatter plots of transmission probabilities reconstructed by DMP algorithm for $M = 10^6$ versus true couplings for a tree (inset of the left figure) and for a power-law network (inset of the right figure).

Fig. 2(b): as demonstrated in the inset of of the Fig. 2(b), an error for large M is due to the inaccuracy of the DMP predictions for the couplings in the vicinity of short loops, while the majority of parameters are correctly predicted by the algorithm.

The Fig. 3 is devoted to the tests in the presence of nodes with hidden information. Because of the large convergence times for MLE and HTS in the case of incomplete information, we were forced to perform tests on small and loopy networks. Since the objective function landscapes naturally contain some local minima for loopy graphs, and in order to reinforce the convergence of algorithms towards the true solution, we choose the initial conditions for the couplings using the following observation: since MLE is supposed to give a good estimation of the parameters for "complete" neighborhoods without hidden nodes, we first estimate this part of couplings using a fast local maximization of (63), and "freeze" these values in all the algorithms. Other couplings are initialized with an upper-bounded estimation of the couplings given by (63) by excluding the hidden nodes. We denote the corresponding modification of the DMP algorithm as DMP+MLE. Note that if the hidden node is a leaf of the network, then no algorithm can reconstruct the transmission probability associated with the ingoing directed edge adjacent to this nodes, therefore, we do not include the such parameters in the computation of the mean error throughout the Fig. 3. As shown in the Fig. 3, (a) and (b), the DMP algorithm shows a robust behavior with respect to the increasing number of hidden nodes, demonstrating comparable results at a substantially lower computational cost on a small power-law network. Finally, the Fig. 3 (d) shows reconstruction results on a real-world Zachary's karate club network [45] with a very large number of small loops. Other examples illustrating the reconstruction performance in the case of incomplete observations in time and in the presence of noise, both on synthetic and real-world networks are provided in the Supplementary Information [43].

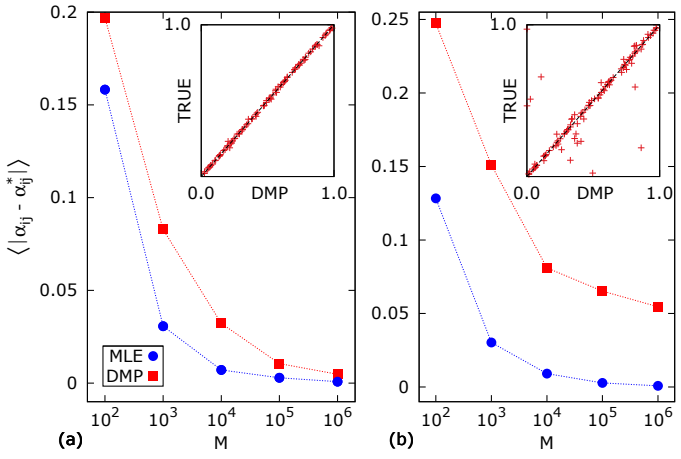

Fig. 2. (Color online) Main figures: Comparison of mean error on the reconstructed couplings as a function of the number of fully observed cascades M for (a) a tree network with $N = 50$ and (b) a connected component of a power-law network with $N = 53$. Insets: Scatter plots of transmission probabilities reconstructed by DMP algorithm for $M = 10^6$ versus true couplings for (a) a tree and (b) a power-law network. The data has been generated with $\{\alpha_{ij}\}_{(ij)}$ uniformly distributed in the range $[0, 1]$, and $T = 10$.

E. Conclusion and perspectives:

An approximate solution given by the DMP equations for spreading processes allowed us to introduce a fast and efficient algorithm for the reconstruction of the transmission probabilities in the presence of hidden information. Contrary to the methods based on the maximization of incomplete likelihood, it can be used for large networks, providing the best performance in the case of sparse networks. Let us indicate some possible generalizations and perspectives. Along with applications to a range of other spreading models [41], the DMP algorithm can be straightforwardly generalized to the case of continuous-time models using the continuous version of the DMP equations [46] (see Appendix B-B) and to dynamically-varying networks in the spirit of [37]: the dynamics of the network can be directly encoded into the DMP equations via the time-dependent couplings $\alpha_{ij}(t)$. An interesting future direction would be to understand if the DMP approach could be used for the network structure learning if some part of it is known, or under some restrictions on the class of networks one tries to reconstruct, e.g. using a ℓ^1-regularization [35], [47]. Some of these directions are further discussed in [43].

IV. Learning Ising Models: new efficient algorithms

This section is a work in progress with M. Chertkov and H. Jang.

A. Preliminaries

Undirected GM, or *Markov random fields*, are widely used in a variety of domains, including image processing, statistical physics, sociology and finance, to model and analyze large-scale systems of interacting elements. The estimation of the best model that fits the experimental data has become a problem of major interest in the last two decades. Despite numerous works and an increase in our computational capability, the problem remains difficult: a generic approach consists in matching a set of observables estimated from data with the ones infered by the model, which is known to be hard in general. The learning problem is often called *inverse problem* in the statistical physics community, by contrast to the "direct" problem of inference.

The Ising model corresponds to a Markov random field with binary variables and pairwise interactions. It is often used as a basic starting point to test methods and a minimal model to analyze correlations between a set of agents. Chow and Liu [48] were the first to adress this problem in 1968 and gave a greedy and exact algorithm to learn tree graphs. Since then, numerous papers have suggested heuristics and exact algorithms for more general settings and elucidated some of the difficulties to learn general Ising models. One of them is the presence of long-range correlations: a node can be more correlated to a distant node than a neighboring node. Decay of correlation, which captures the intuition of asymptotic independence between two spins when their graph-distance increases, was long believed to be an unavoidable feature for a model to be efficiently learned [49]. However, Guy Bresler [5] showed this year that one can learn arbitrary bounded-degree graph without any assumption on the interactions, with roughly the same complexity as the tree-case. His algorithm is yet unpractical and we suggest further works based on objective-optimization which might present several advantages for practical implementation: computational efficiency and direct generalization. Our original contribution is a new convex estimator and a spin-flip regularization.

In Section IV-A1 we introduce the inverse Ising model problem and discuss its limitations. In Section IV-B we give a short review of the past work on learning bounded-degree Ising models. In Section IV-C we present our new annealed estimator which is compared to the existing quenched pseudo-likelihood, and the spin-flip regularization, as an alternative to the lasso regularization. Finally we discuss about future works and possible new settings in Section IV-D.

1) Ising Model: We consider an Ising model on a graph $G = (V, E)$ with $|V| = N$. A binary random variable $\Sigma_i \in \{-1, +1\}$ (spin) is associated to each vertex $i \in V$. The joint distribution of the spins is given by:

$$P_{J,h}(\sigma) = \frac{1}{Z(J)} \exp\left(\sum_{(i,j)\in E} J_{ij}\sigma_i\sigma_j + \sum_{i\in V} h_i\sigma_i \right) \quad (72)$$

The distribution is then parametrized by $\{\{J_{ij}\}_{(i,j)\in E}, \{h_i\}_{i\in V}\} \in \mathbb{R}^{|E|} \times \mathbb{R}^{|V|}$. For simplicity, we will reparametrize the model with $\{J_{ij}\}_{(i,j)\in V\times V}$ where $J_{ij} \neq 0$ if and only if $(i, j) \in E$.

We will denote $\alpha = \min_{(i,j)\in E} |J_{ij}| > 0$, $\beta = \max_{(i,j)\in E} |J_{ij}|$, $H = \max_{i\in V} |h_i|$ and $\partial i = \{j | j \in V, (i, j) \in E\}$ the neighborhood of the node i.

2) Learning of Graphical Models: given a subset $\mathcal{G}_{N,\text{sub}} \subset \mathcal{G}_N$ of the set of graphs with N nodes, we consider the set of Ising models on a graph $G \in \mathcal{G}_{N,\text{sub}}$ and with coefficients in

$$\begin{aligned} \Theta_{\alpha,\beta,H}(G) \equiv \big\{ & J \in \mathbb{R}^{N(N-1)/2}, h \in \mathbb{R}^N \mid |h_i| \leq H, \\ & \alpha \leq |J_{ij}| \leq \beta \text{ if } (i,j) \in E \\ & J_{ij} = 0 \text{ otherwise} \big\} \end{aligned} \quad (73)$$

The learning problem consists in reconstructing $\{J, h\}$, given a collection $\Sigma^M = \Sigma^{(1)}, \ldots, \Sigma^{(M)}$ of M independent and identically distributed samples drawn from an Ising Model $\{J, h\}$. We remark that if the underlying graph is given, it is relatively easy to reconstruct the coefficients:

$$\begin{aligned} \exp(2h_i) &= \frac{P(\sigma_i|\sigma_{\partial i})}{P(-\sigma_i|-\sigma_{\partial i})} \\ \exp(2J_{ij}) &= \frac{P(\sigma_i|\sigma_{\partial i\setminus j}, \sigma_j)}{P(\sigma_i|\sigma_{\partial i\setminus j}, -\sigma_j)} \end{aligned}$$

Hence, we will focus on the problem of *structure learning*. An algorithm of structure learning is a map, also called *decoder*:

$$\Phi : (\{-1, +1\}^N)^M \to \mathcal{G}_{N,\text{sub}} \quad (74)$$

which maps the input samples to an undirected graph $\widehat{G} = \Phi(\Sigma^M)$. We will consider the following maximum-risk measure to study the performance of the decoder:

$$\sup_{G\in\mathcal{G}_{N,\text{sub}}} \sup_{\{J,h\}\in\Theta_{\alpha,\beta,H}(G)} P_{\{J,h\}}(\Phi(\Sigma^M) \neq G) \quad (75)$$

which corresponds to the error probability of the worst case in the subset $\mathcal{G}_{N,\text{sub}}$.

We define the sample complexity of the decoder as follows, given a small parameter $\delta > 0$:

$$\begin{aligned} M_\Phi(\delta) = & \sup_{G\in\mathcal{G}_{N,\text{sub}}} \sup_{\{J,h\}\in\Theta_{\alpha,\beta,H}(G)} \\ & \inf\{n \in \mathbb{N} | P_{\{J,h\}}(\Phi(\Sigma^n) = G) \geq 1 - \delta\} \end{aligned} \quad (76)$$

The task is then to find an efficient decoder that has a good sample complexity, i.e. that minimizes the worst case error for a given number of samples.

The Ising model is in the exponential family and is therefore completely characterized by its first and second moments:

$$\nu_i = \mathbb{E}_P[\sigma_i], \qquad \mu_{ij} = \mathbb{E}_P[\sigma_i\sigma_j] \quad (77)$$

called *sufficient statistics* of the distriution. However, it does not provide any clue on how to define a tractable decoder that matches the first and second moments.

The search for a practical decoder corresponds to finding the best trade-off between the sample complexity and the numerical complexity.

3) Information theoretic limitations: Santhanan and Wainwright gave in [50] theoretical bounds on the sample complexity using Fano's lemma and showed that these bounds can be achieved with infinite computational power. Given an arbitrary decoder which selects among a family of L Ising Model $\{G^{(1)},\dots,G^{(L)}\}$ and a collection Σ^M of M samples drawn i.i.d from the Ising Model $G^{(k)}$, the lemma states:

Lemma 6. *For a given $\epsilon > 0$, if the sample size M verifies:*

$$M < (1-\epsilon)\frac{\log(L)}{I(\Sigma^1,k)},\ \textit{or}\ M < (1-\epsilon)\frac{L^2\log(L)}{2\sum_{l<m} D(G^{(l)},G^{(m)})}$$

where $I(\Sigma^1,k)$ is the mutual information between one sample and the index of the model, and $D(G^{(l)},G^{(m)})$ is the symmetrized Kullback-Leiber (KL) divergence between the two distributions $G^{(l)}$ and $G^{(m)}$, then

$$\max_{k=1,\dots,L} P_{G^{(k)}}(\Phi(\Sigma^M)\neq G^{(k)}) \geq \epsilon - \frac{1}{\log(L)}$$

Therefore, to have a probability of error smaller than $\epsilon - \frac{1}{\log(L)}$, M must be larger than the two bounds of the lemma. Therefore Fano's lemma gives a lower bound on the sample complexity.

By using this lemma with a proper choice of the sub-family, one can show that if we consider the set $\mathcal{G}_{N,d}$ of all graphs with a bounded-degree d:

$$M_\Phi(\delta) \leq (1-\delta)\frac{e^{\beta d}\log(\frac{pd}{4}-1)}{4\alpha d e^{\alpha}}. \tag{78}$$

This bound shows that, in the general case ($d = N$), with this measure of performance, one cannot avoid an exponential in N sample complexity. One then need to restrict the class of models we considers.

In the rest of this work, we will focus on learning bounded-degree graphs, although our annealed estimator is not specific to this setting.

B. Short review of the previous works

The problem of structure learning was first adressed by Chow and Liu in [48] in which they presented an exact algorithm to learn tree-graphs with a runtime complexity of $O(N^2)$. Using a pairwise factorization of the distribution specific to the tree-structure, they showed that by minimizing the KL divergence between the data and tree-graph distributions, learning the graph is equivalent to finding the maximum spanning tree, where the weight between two nodes corresponds to their mutual information. Since then, numerous works have considered other restrictions. We can distinguish between two types of restrictions: on the structure and on the interactions.

For the first case, Bresler and al. [51] have considered the family of bounded-degree graphs. By noticing the following structural property of independency: for $i \in V$ and $j \notin \partial i$

$$P(\sigma_i|\sigma_{\partial i}) = P(\sigma_i|\sigma_{\partial i},\sigma_j) \tag{79}$$

they suggested an exhaustive search of the neighborhood. For each of the $\binom{N}{d}$ possible neighborhoods, one check the conditional independence for each of the other nodes. They showed that their algorithm achieves the theoretical bound of sample complexity $O(\log N)$. However, for each sample, one need to test N times for each possible neighborhood of each node, wich gives an overall runtime complexity of $O(N^{d+2}\log N)$.

For the second type of restrictions, it was observed in [51] that efficient learning is possible if one assumes decay of correlation. Other articles have assumed decay of correlation to give theoretical guarantees (e.g. [52], [53]). Bento and Montanari [49] even showed that the pseudo-likelihood with a $l1$-regularization fails with high probability for regular ferromagnetic Ising models when there is no more decay of correlation. Until the recent paper [54], all known efficient algorithms were assuming decay of correlation and it was long believed that it was an unavoidable property for efficient learning [49].

Bresler et al. [54] presented an algorithm to efficiently learn antiferromagnetic models with strong interactions (no decay of correlation). Their algorithm is exploiting a specific feature of the probability that characterizes the presence of an edge in the case of strong repelling interactions. Then, one year later, Bresler [5] presented an efficient algorithm for arbitrary d-bounded-degree graphs and arbitrary couplings. He showed that there exists a subtle and non-intuitive structural property of the Ising model that can reduce the size of the potential neighborhood: for each node i, one can recursively construct by using mutual information a neighborhood that contains the true neighborhood and whiose size is bounded and do not depend on N, the total number of spins. Hence, one can find these potential neighborhoods and exploit the conditional independency (79) on these subsets. The resulting algorithm has a sample complexity of $O(e^{e^{cd}}\log N)$ and a runtime complexity of $O(e^{e^{cd}}N^2\log N)$. Therefore, his algorithm has roughly the same complexity in N as for the tree-case. However, the double-exponential in d prefactor makes this algorithm unpractical.

C. Learning as an optimization problem

Despite achieving the theoretical bound $O(\log N)$ on the sample complexity and an overall runtime complexity of $O(N^2)$, Guy Bresler's algorithm remains computationally too heavy for real-life applications, due to the double exponential in d of the prefactor. Furthermore, his algorithm seems difficult to implement for a non-binary alphabet and higher-order interactions. In this section, we follow an alternative approach to the greedy algorithms: optimization, i.e. the estimator is defined as the optimum of a function.

1) Pseudo-likelihood: In the bayesian framework, a common learning method relies on the the maximization of the likelihood of the data $\Sigma^M = \{\Sigma_{(1)}, \dots, \Sigma^{(M)}\}$:

$$\begin{aligned}\{J,h\} &= \arg\max_{\{J,h\}} \log \prod_{i=1}^{M} P(\Sigma^{(i)} \mid \{J,h\}) \\ &= \arg\max_{\{J,h\}} \langle \log P(\Sigma \mid \{J,h\}) \\ &= \arg\max_{\{J,h\}} \sum_{(i,j)\in E} J_{ij}\langle\sigma_i\sigma_j\rangle + \sum_{i\in V} h_i\langle\sigma_i\rangle \\ &\quad - \log Z(\{J,h\}) \end{aligned} \tag{80}$$

This optimization is convex but requires to have an efficient algorithm to solve the (direct) inference problem of computing the partition function $Z(\{J,h\})$ which is non-tractable in the general case. Even if the direct problem is tractable, e.g. the planar case, one still need to test a super-exponential number of graphs.

An alternative estimator has been suggested in the statistics litterature in the seventies, called *maximum pseudo-likelihood*, because local likelihood are used as proxies to the global likelihood, avoiding the computation of the partition function. For each node $i \in V$, the maximum pseudo-likelihood is defined by:

$$\{J_{\partial i}, h_i\} = \arg\max_{\{J_{\partial i}, h_i\}} \frac{1}{M} \log \prod_{k=1}^{M} P_i(\sigma_i^{(k)} \mid \sigma_{\partial i}^{(k)}) \tag{81}$$

where $P_i(\sigma_i|\sigma_{\partial i})$ is the marginal probability of σ_i given its neighborhood:

$$P_i(\sigma_i|\sigma_{\partial i}) = \frac{e^{\sigma_i\left[\sum_{j\in\partial i} J_{ij}\sigma_j + h_i\right]}}{e^{\sum_{j\in\partial i} J_{ij}\sigma_j + h_i} + e^{-\sum_{j\in\partial i} J_{ij}\sigma_j - h_i}} \tag{82}$$

The function to maximize is hence equal to

$$\begin{aligned} L_i(\Sigma^M, J_{\partial i}, h_i) &= h_i\langle\sigma_i\rangle + \sum_{j\in\partial i} J_{ij}\langle\sigma_i\sigma_j\rangle \\ -\langle \log(e^{\sum_{j\in\partial i} J_{ij}\sigma_j + h_i} &+ e^{-\sum_{j\in\partial i} J_{ij}\sigma_j - h_i}\rangle. \end{aligned} \tag{83}$$

In the following, we will call L_i the quenched pseudo-likelihood in order to contrast it with the "annealed" estimator that we will introduce below.

By taking the first derivative and replacing in (83) the average on the data $\langle\cdot\rangle$ by the exact average $\mathbb{E}_{J_{\partial i}, h_i}[\cdot]$, the true couplings are an optimum of the pseudo-likelihood. The estimor is therefore consistent and gives the right coefficients. Furthermore, we remark that for any distribution in the exponential family, (81) is convex. The pseudo-likelihood is therefore a convex estimator with a unique maximum equal to the right coefficients when $M \to +\infty$. Therefore, one can use the optimization locally and sequentially, i.e. $\forall i \in \mathcal{V}$, to recover J and h but also the structure of the graph (for example, by removing the edges with a coupling inferior to a certain treshold). However, as we saw in IV-A3, if the bare optimization of the quenched pseudo-likelihood is applied at finite M, the decoder will have an exponential sample complexity in the number of nodes, and hence, an exponential runtime complexity.

To overcome that problem, adding an $l1$-regularization local term to (83) was suggested in [55]:

$$\begin{aligned} &\max_{\{J_{\partial i}, h_i\}} L_i(\Sigma^M, J_{\partial i}, h_i) + \lambda l_1(J_{\partial i}) = \\ &\max_{\{J_{\partial i}, h_i\}} L_i(\Sigma^M, J_{\partial i}, h_i) + \lambda \sum_{j\in\partial i} |J_{ij}|. \end{aligned} \tag{84}$$

This regularization, also called *lasso regularization*, has been widely used in optimization to impose sparsity to the reconstructed parameters. Ravikumar et al. showed that under some assumptions on the underlying model, one can reconstruct the true graph with a sample complexity of $O(d^3 \log N)$.

The assumptions are on the hessian H of the pseudo-likelihood at the right couplings with infinite sampling. The *dependency condition* is that the sub-matrix H_{SS} of the Hessian with indices in the neighborhood S has bounded eigenvalues Λ:

$$C \geq \Lambda_{\max}(H_{SS}) \geq \Lambda_{\min}(H_{SS}) \geq c > 0. \tag{85}$$

The *incoherence condition* (the rest of the graph cannot have an overly strong effect on the neighborhood) is stated as follows: there exists $0 < \alpha \leq 1$ such that

$$\|H_{S^cS} H_{SS}^{-1}\|_\infty \leq 1 - \alpha. \tag{86}$$

These conditions are not easily linked to the coefficient but are believed to be of the decay of correlation type. Bento and Montanari showed in [49] that they are indeed quite restrictive: the lasso regularization fails with high probability for a uniforml random graph of regular degree $d > 3$ when the temperature is below a treshold (related but not equal to the critical temperature).

However, the regularized pseudo-likelihood is widely used in practice due to its versatility, simplicity to implement, and computational efficience. Despite its theoretical limitations for exact reconstruction when using the worst case measure, a number of works [56] have shown that this estimator is in average efficient for general topologies, given a fixed number of samples.

2) Screening objective: Due to their great practicality, it is interesting to explore other convex estimators. Can we extend or modify the subspace of models that can be efficiently learned with an optimization ? Can we beat the decay of correlation with a proper choice of objective function ?

We consider the following function for each node $i \in V$:

$$S_i(\Sigma^M, J_{\partial i}, h_i) = -\log\langle e^{-\sigma_i[h_i + \sigma_i \sum_{j\in\partial i} J_{ij}\sigma_j]}\rangle. \tag{87}$$

First we remark that if we take the exact averaging and

evaluate the function at the right couplings:

$$\begin{aligned}&\langle e^{-\sigma_i[h_i^*+\sigma_i\sum_{j\in\partial i}J_{ij}^*\sigma_j]}\rangle\\&=\frac{1}{Z(J^*,h^*)}\sum_\sigma\exp\big(\sum_{k\in V}h_k^*\sigma_k\\&+\sum_{(k,l)\in E}J_{kl}^*\sigma_k\sigma_l-\sigma_i[h_i^*+\sigma_i\sum_{j\in\partial i}J_{ij}^*\sigma_j]\big)\\&=\frac{2Z(J_{\setminus i}^*,h_{\setminus i}^*)}{Z(J^*,h^*)}\end{aligned}\tag{88}$$

where $Z(J_{\setminus i}^*,h_{\setminus i}^*)$ is the partition function of the Ising model where the node i has been erased. The function "screens" the interaction between a node and the rest of the graph. Therefore we will call (87) *screening ojective*.

Secondly, if the distribution is in the exponential family, (87) is convex. Let us compute the derivative at the right couplings for the exact averaging:

$$\partial_{J_{ik}}S_i(\Sigma^\infty,J_{\partial i}^*,h_i^*)=\mathbb{E}_{|J_{\partial_i}=0,h_i=0}[\sigma_i\sigma_k]=0\tag{89}$$

$$\partial_{h_i}S_i(\Sigma^\infty,J_{\partial i}^*,h_i^*)=\mathbb{E}_{|J_{\partial_i}=0,h_i=0}[\sigma_i]=0\tag{90}$$

Therefore, we have an objective which is convex and which minimum corresponds to the the right couplings when $M\to+\infty$.

We define the following estimator:

$$\{J_{\partial i},h_i\}=\arg\min_{\{J_{\partial i},h_i\}}-\log\langle e^{-\sigma_i[h_i+\sigma_i\sum_{j\in\partial i}J_{ij}\sigma_j]}\rangle\tag{91}$$

which will be called *annealed screening*, by contrast to the quenched pseudo-likelihood.

In the same way as in IV-C1, one can add an $l1$-regularization. The exact Hessian at the minimum is given by (if we consider only the pairwise parameters):

$$\begin{aligned}H_{k,l}=&\mathbb{E}_{|J_{\partial_i}=0,h_i=0}[\sigma_k\sigma_l]\\&-\mathbb{E}_{|J_{\partial_i}=0,h_i=0}[\sigma_k]\mathbb{E}_{|J_{\partial_i}=0,h_i=0}[\sigma_l]\end{aligned}\tag{92}$$

In the case of a tree, the Hessian will be block-diagonal, with non-zero coefficient only if k and l are on the same branch starting from i. The conditions (85) and (86) will be always satisfied in this case. If a similar result to [55] on the regularized pseudo-likelihood stands for the annealed case, this remark seems to indicate that this estimator do not have a phase transition on the sampling complexity for trees.

Besides, it is interesting to compare the performance of the quenched and annealed objective and see if we can retrieve some of their property they have in the inference setting. We expect that in the case of correlation decay, the two schemes will have comparable performance. Based on what differentiate annealed and quenched in traditional statistical physics context, namely that the optimality of the annealed scheme will not change with the transition from an ordered to a glassy phase, we would also expect a difference in their behavior below the critical temperature. Besides, the annealed functions have a much better, large-deviation type, concentration than their quenched counter-parts, which might turn out advantageous if it used in the context of a biased sampling or an active learning.

This convex estimator is currently investigated numerically.

3) Spin-flip regularization: It might also be interesting to explore other regularization functions to replace/be added to the $l1$-regularization. Here we suggest a regularization based on a specific symmetry of the Ising model: the distribution is invariant with respect to a flip of the spin variables within a subset of nodes and the flip of the sign of their magnetic fields and of the couplings on the boundary.

This symmetry imposes a set of constraints on the reconstructed couplings. Given an Ising model (G,J,h), we consider an embedding of the graph in $\mathbb{R}^3$ such that edges intersects only at their extremities. For any volume $\mathcal{V}\subset\mathbb{R}^3$ such that its surface $\partial\mathcal{V}$ intersects at most one time each edge, and for any partition of its surface into two subsets $\partial\mathcal{V}=\Gamma\cup\Gamma^c$, we have the following equality :

$$\begin{aligned}\mathbb{E}_{J,h}\left[\prod_{(i,j)\perp\Gamma}\exp(-2J_{ij}\sigma_i\sigma_j)\prod_{i\in\mathcal{V}}\exp(-2h_i\sigma_i)\right]\\=\mathbb{E}_{J,h}\left[\prod_{(i,j)\perp\Gamma^c}\exp(-2J_{ij}\sigma_i\sigma_j)\right]\end{aligned}\tag{93}$$

The proof of this relation is simple: one writes explicitly the expectation and notices that by flipping the spins inside $\mathcal{V}$ one obtains the equality.

The simplest equality that we can use to regularize our objective function is to consider the following local constraints: we consider for each node $i\in V$, a volume $\mathcal{V}$ containing only i and $\Gamma\equiv\partial\mathcal{V}$

$$\forall i\in\mathcal{V}:\quad\mathbb{E}_{J,h}\left[\exp\left(-2h_i\sigma_i-2\sum_{j\in V}J_{ij}\sigma_i\sigma_j\right)\right]=1,\tag{94}$$

which follows explicitly from the $\sigma_i\to-\sigma_i$ change of variables within the expectation. Here in (94) we assume that $J_{ij}=0$ if $(i,j)\notin E$.

We suggest to try using the constraint (94), or similar constraints, as a substitute for the $l1$-regularization, in the annealed and pseudo-likelihood optimization schemes. Specifically, one may have the following (properly weighted) addition to the cost in (83) or in (87)

$$\left(\log\mathbb{E}_{J,h}\left[\exp\left(-2h_i\sigma_i-2\sum_{j\in V}J_{ij}\sigma_i\sigma_j\right)\right]\right)^2\tag{95}$$

which is differentiable. We notice that this regularization is not convex: it has at least two minima, the right coefficients and the trivial coefficients $h_i=0$ and $J_{ij}=0$. However, if we manage to impose a local domain restriction in the space of J_{ij} and h_i (e.g. by starting the optimization without regularization), one may ensure local convexity of the spin-flip regularizer. This regularization might, by enforcing an additional constraint, speed up the optimization (deeper well at the minimum) and improve the result in different contexts (in the case of a small number of data).

D. Discussion

Convex optimization represents a simple-to-implement, general-scope framework widely used in applications. Here we suggest to explore this approach in the context of Ising Model learning. We saw in IV-C1, that efficient learning is restricted to a subspace of the coefficients in the case of pseudo-likelihood. Can we extend this subspace with other objective functions? Then, one might try different functions or even mix them in order to learn the model. An other important question concerns the decay of correlation: does optimization require decay of correlation for an efficient learning? Besides, the maximum-risk measure in the case of exact learning is too strong for practical application with M finite: it might be interesting to consider other measures of performance (Kullback-Leiber divergence, average error, etc.).

Bresler's algorithm uses a particular structural property of the Ising model. Following this idea, we introduced in IV-C3 a regularizer exploiting the spin-flip symmetry. Can we mix the greedy and optimization approach? For example, one might sequentially optimize the pseudo-likelihood, tresholding the coefficents to reduce the potential neighborhood and then switch to a greedy approach.

The use of the annealed screening might also be interesting in other contexts, e.g. the samples are biased (they have been produced by a markov process). It might also present some advantages if an *active learning* is developed to speed up the learning process, or to improve the use of the data.

APPENDIX A
PROOFS OF LEMMAS

A. Proof of Lemma 4

Proof: We prove Lemma 4 only for the BSC (the proof for the BEC is almost identical). For a given ρ and p, let us define the following function

$$\begin{aligned} g^{\text{BSC}}\left(x_0, x_c, \underline{y}, \eta\right) &= f\left(x_0, x_c, \underline{y}, \rho\right) + k\left(x_0, x_c, \rho, p\right) \\ &- 2\eta\left(x_0\left(1-\rho\right) + x_c\rho\right). \end{aligned} \tag{96}$$

The function $g^{\text{BSC}}\left(x_0, x_c, \underline{y}, \eta\right)$ corresponds to the exponent of the loop series (42) associated with the loop type $(x_0, x_c, \underline{y})$. In order to prove Lemma 4, we have to find $\widetilde{\eta} < 0$ such that g^{BSC} is non-positive on $D(\rho) \times [\widetilde{\eta}, +\infty[$.

We first show that for any $\widetilde{\eta}_1 < 0$, there exists a neighborhood U of $(x_o, x_c, \underline{y}) = (0,0,0)$ such that g^{BSC} is non-positive on $U \cap D(\rho) \times [\widetilde{\eta}_1, +\infty[$. For a fixed $\widetilde{\eta}_1 < 0$ we construct a function $\overline{g}^{\text{BSC}}$ that is an upper bound of g^{BSC}. We restrict ourselves to the domain $V \cap D(\rho) \times [\widetilde{\eta}_1, +\infty[$, where $V = \mathbb{B}(0, 1/3r)$ is the ball of radius $1/3r$ centered at $(0,0,0)$.

Let us explicitly write down the function (96) term by term

$$\begin{aligned} g^{\text{BSC}}\left(x_0, x_c, \underline{y}, \eta\right) &= -2\eta\left(x_0\left(1-\rho\right) + x_c\rho\right) \\ &+ \left(\rho x_c - \left(1-\rho\right)x_0\right)\ln\left(\frac{1-p}{p}\right) \\ &+ \frac{l}{r}\sum_{t=1}^{\lfloor r/2\rfloor} y_t \ln\left(\begin{array}{c} r \\ 2t \end{array}\right) \\ &- l h_2\left(\left(1-\rho\right)x_0 + \rho x_c\right) \\ &+ \left(1-\rho\right)h_2\left(x_0\right) + \rho h_2\left(x_c\right) \\ &- \frac{l}{r}\left(1 - \sum_{t=1}^{r} y_t\right)\ln\left(1 - \sum_{t=1}^{r} y_t\right) \\ &- \frac{l}{r}\sum_{t=1}^{r} y_t \ln y_t. \end{aligned} \tag{97}$$

We bound each term of (97) separately. Denote the fraction of variable nodes in the loop by $X = x_0\left(1-\rho\right) + x_c\rho$. The inequalities below trivially hold

$$\begin{aligned} \left(\rho x_c - \left(1-\rho\right)x_0\right)\ln\left(\frac{1-p}{p}\right) &\leq 2\ln\left(\frac{1-p}{p}\right)X \\ \frac{l}{r}\sum_{t=1}^{\lfloor r/2\rfloor} y_t \ln\left(\begin{array}{c} r \\ 2t \end{array}\right) &\leq l\ln\left(\begin{array}{c} r \\ 2\lfloor r/2\rfloor \end{array}\right)X \\ -2\eta\left(x_0\left(1-\rho\right) + x_c\rho\right) &\leq -2\widetilde{\eta}_1 X. \end{aligned} \tag{98}$$

As the entropy is a concave function, we have the following inequality

$$\left(1-\rho\right)h_2\left(x_0\right) + \rho h_2\left(x_c\right) \leq h_2(X). \tag{99}$$

Concativty of $-x\ln x$ gives us

$$\begin{aligned} -\sum_{t=1}^{\lfloor r/2\rfloor} y_t \ln y_t &\leq -\left(\sum_{t=1}^{\lfloor r/2\rfloor} y_t\right)\ln\left(\frac{1}{\lfloor r/2\rfloor}\sum_{t=1}^{\lfloor r/2\rfloor} y_t\right) \\ &\leq -\left(\sum_{t=1}^{\lfloor r/2\rfloor} y_t\right)\ln\left(\sum_{t=1}^{\lfloor r/2\rfloor} y_t\right) \\ &\quad + r\ln\left(\lfloor r/2\rfloor\right)X. \end{aligned} \tag{100}$$

Note that since the domain is restricted to types in a ball of radius $1/3r$, the fraction of variable nodes in a loop is upper-bounded $X \leq 1/3r$. In particular it implies that

$$\begin{aligned} \sum_{t=1}^{\lfloor r/2\rfloor} y_t &\leq \frac{r}{2}\left(\sum_{t=1}^{\lfloor r/2\rfloor}\frac{2t}{r}y_t\right) \\ &= \frac{r}{2}X \\ &\leq \frac{1}{6} \\ &\leq \frac{1}{e}, \end{aligned} \tag{101}$$

where e is the Euler constant. Finally as the entropy is

increasing on $\left[0, \frac{1}{e}\right]$, we have

$$\frac{l}{r} h_2 \left(\sum_{t=1}^{\lfloor r/2 \rfloor} y_t \right) \leq \frac{l}{r} h_2 \left(\frac{r}{2} X \right). \quad (102)$$

The upper bound on the function (97) is simply the sum of Inequalities (98), (99), (100) and depends only on the fraction of variable nodes in a loop i.e. $\overline{g}^{\text{BSC}}\left(x_0, x_c, \underline{y}, \eta\right) \equiv \overline{g}^{\text{BSC}}(X)$

$$\overline{g}^{\text{BSC}}(X) = \frac{l}{r} h_2 \left(\frac{r}{2} X \right) - (l-1) h_2 (X) + MX \quad (103)$$

where M is a constant independent of η and ρ

$$M = 2 \ln \left(\frac{1-p}{p} \right) + l \ln \left(\begin{array}{c} r \\ 2\lfloor r/2 \rfloor \end{array} \right) + l \ln \left(\lfloor r/2 \rfloor \right) - 2\widetilde{\eta}_1. \quad (104)$$

Notice that $\overline{g}^{\text{BSC}}(0) = 0$ and that the derivative $\frac{d}{dX} \overline{g}^{\text{BSC}}(X)$ behaves like $\left(\frac{l}{2} - 1\right) \ln X$ in the neighborhood of 0. Hence, for $l \geq 3$, there exists $\delta > 0$ such that $\overline{g}^{\text{BSC}}$ is negative on $]0, \delta]$. Therefore for all types $(x_o, x_c, \underline{y}) \in D(\rho)$ in the domain $U = \mathbb{B}(0, \delta) \cap \mathbb{B}(0, 1/3r)$ and for all $\eta \in [\widetilde{\eta}_1, +\infty[$ we have

$$\begin{aligned} g^{\text{BSC}}\left(x_0, x_c, \underline{y}, \eta\right) &\leq \overline{g}^{\text{BSC}}(X) \\ &\leq 0. \end{aligned} \quad (105)$$

By hypothesis the maximum of (48) is uniquely achieved in $(0, 0, 0)$ for $\eta = 0$. It implies that there exists $\lambda < 0$ such that

$$\max_{\left(x_0, x_c, \underline{y}\right) \in D(\rho) \setminus U} f\left(x_0, x_c, \underline{y}, \rho\right) + k\left(x_0, x_c, \rho, p\right) = \lambda. \quad (106)$$

Therefore for $\eta > \widetilde{\eta}_2 = \lambda/2$

$$\max_{\left(x_0, x_c, \underline{y}\right) \in D(\rho) \setminus U} g^{\text{BSC}}\left(x_0, x_c, \underline{y}, \eta\right) \leq \lambda - 2\widetilde{\eta}_2 = 0. \quad (107)$$

We see that $\widetilde{\eta} = \max(\widetilde{\eta}_1, \widetilde{\eta}_2) < 0$ satisfies by construction the condition of Lemma 4. ∎

B. Proof of Lemma 5

Proof: We prove Lemma 5 only for the BSC (the proof for the BEC is almost identical). For a given ρ and p, we recall the function $g^{\text{BSC}}_{p,\rho} \equiv g^{\text{BSC}}$ and $\overline{g}^{\text{BSC}}_{p,\rho} \equiv \overline{g}^{\text{BSC}}$ as defined in Appendix A-A. We prove that for n sufficiently large and for all $\delta \in \left[-\sqrt{n^{-1} \ln n}, \sqrt{n^{-1} \ln n}\right]$, the function $g^{\text{BSC}}_{p,\rho+\delta}$ is still non-positive on $D(\rho)$.

First notice that the upper bound $\overline{g}^{\text{BSC}}_{p,\rho}$ does not depend on ρ. Using the same argument as in Appendix A-A, there exists a neighborhood U of $(0, 0, 0)$ such that for all type $(x_0, x_c, \underline{y}) \in U \cap D(\rho + \delta)$ and for all $\delta \in \left[-\sqrt{n^{-1} \ln n}, \sqrt{n^{-1} \ln n}\right]$

$$\begin{aligned} g^{\text{BSC}}_{p,\rho+\delta}\left(x_0, x_c, \underline{y}, 0\right) &\leq \overline{g}^{\text{BSC}}_{p,\rho}(X) \\ &\leq 0. \end{aligned} \quad (108)$$

It remains to show that the variation of g^{BSC} on $D\left(\rho + \delta\right) \setminus U$ is bounded. Let us make the change of variables $(x_0, x_c) \to (X, x_c)$ and $g^{\text{BSC}}_{p,\rho+\delta}\left(x_0, x_c, \underline{y}, 0\right) \to g^{\text{BSC}}_{p,\rho+\delta}\left(X, x_c, \underline{y}, 0\right)$. The following inequality holds

$$\begin{aligned} g^{\text{BSC}}_{p,\rho+\delta}\left(X, x_c, \underline{y}, 0\right) &\leq 2\sqrt{\frac{\ln n}{n}} \left(\ln 2 + \ln \left(\frac{1-p}{p} \right) \right) \\ &+ g^{\text{BSC}}_{p,\rho}\left(X, x_c, \underline{y}, 0\right). \end{aligned} \quad (109)$$

Hence we can bound the maximum of g^{BSC} on $D(\rho + \delta) \setminus U$ by

$$\begin{aligned} \max_{(X, x_c, \underline{y}) \in D(\rho+\delta) \setminus U} & g^{\text{BSC}}_{p,\rho+\delta}\left(X, x_c, \underline{y}, 0\right) \leq \\ \max_{(X, x_c, \underline{y}) \in D(\rho) \setminus U} & g^{\text{BSC}}_{p,\rho}\left(X, x_c, \underline{y}, 0\right) + c\sqrt{\frac{\ln n}{n}} \end{aligned} \quad (110)$$

The maximum of $g^{\text{BSC}}_{p,\rho}\left(X, x_c, \underline{y}, 0\right)$ on $D(\rho + \delta) \setminus U$ is by hypothesis negative (see Equation (106)). Therefore for n sufficiently large we have that for all $\delta \in \left[-\sqrt{n^{-1} \ln n}, \sqrt{n^{-1} \ln n}\right]$

$$\max_{(x_0, x_c, \underline{y}) \in D(\rho+\delta) \setminus U} g^{\text{BSC}}_{p,\rho+\delta}\left(x_0, x_c, \underline{y}, 0\right) \leq 0, \quad (111)$$

which concludes the proof. ∎

APPENDIX B
LEARNING SPREADING PROCESS: SUPPLEMENTAL MATERIAL

A. Computation of the gradient in the DMP algorithm

In this section, we will provide details for the derivation of the dynamic message-passing equations that we use to compute the gradient of the costfunction $J(t)$ (equation (71) in the main text) at each iteration step of the DMP algorithm:

$$-\frac{\partial J(t)}{\partial \alpha_{rs}} = \sum_{i \in \mathcal{O}} [m^i_*(t) - m^i(t)] \frac{\partial m^i(t)}{\partial \alpha_{rs}}. \quad (112)$$

Using the equation (69) of the main text, we have

$$\begin{aligned} \frac{\partial m^i(t)}{\partial \alpha_{rs}} = P^i_S(0) \big[& \sum_{k \in \partial i} \frac{\partial \theta^{k \to i}(t-1)}{\partial \alpha_{rs}} \prod_{l \in \partial i \setminus k} \theta^{l \to i}(t-1) \\ & - \sum_{k \in \partial i} \frac{\partial \theta^{k \to i}(t)}{\partial \alpha_{rs}} \prod_{l \in \partial i \setminus k} \theta^{l \to i}(t) \big]. \end{aligned} \quad (113)$$

Let us introduce useful notations:

$$\frac{\partial \theta^{k \to i}(t)}{\partial \alpha_{rs}} \equiv p^{k \to i}_{rs}(t), \qquad \frac{\partial \phi^{k \to i}(t)}{\partial \alpha_{rs}} \equiv q^{k \to i}_{rs}(t). \quad (114)$$

Then, since the initial dynamic messages $\{\theta^{i \to j}(0)\}_{(ij) \in E}$ and $\{\phi^{i \to j}(0)\}_{(ij) \in E}$ are independent on the couplings, we have $p^{k \to i}_{rs}(0) = q^{k \to i}_{rs}(0) = 0$ for all k, i, r and s, and these quantities can be computed iteratively using the analogues of

(69) and (70):

$$p_{rs}^{k\to i}(t) = p_{rs}^{k\to i}(t-1) - \alpha_{ki} q_{rs}^{k\to i}(t-1) - \phi^{k\to i}(t-1)\mathbb{1}[k=r, i=s], \quad (115)$$

$$q_{rs}^{k\to i}(t) = (1-\alpha_{ki}) q_{rs}^{k\to i}(t-1) - \phi^{k\to i}(t-1)\mathbb{1}[k=r, i=s] + P_S^k(0) \sum_{l\in\partial k\setminus i} p_{rs}^{l\to k}(t-1) \prod_{n\in\partial k\setminus\{i,l\}} \theta^{n\to k}(t-1) - P_S^k(0) \sum_{l\in k\setminus i} p_{rs}^{l\to k}(t) \prod_{n\in\partial k\setminus\{i,l\}} \theta^{n\to k}(t). \quad (116)$$

Hence, at each time step, we compute the marginals using the equations (68)-(70) and use (113)-(116) to update the couplings. In principle, at each iteration step of the algorithm we could run the DMP equations for all T time steps with the current estimation of the couplings, and only then update the transmission probabilities using the derivative of the total cost function $J = \sum_{t=0}^{T-1} J(t)$; we found that the "online"-like update at each time step in the spirit of [44] leads to a faster convergence of the algorithm. An intuition behind this choice is as follows: instead of accumulating the error due to the current estimation of the couplings through the whole process, we adjust the couplings progressively as the processes spreads throughout the network.

In practice, we observed that a simple gradient descent with a fixed learning rate ϵ demontsrated good convergence to the optimum, and this is the procedure that we used for producing all the plots in this paper. For most of the plots, we used $\epsilon = 5.0$, and a tolerance on the change of the objective function $\delta = 10^{-12}$ as a stopping criteria for the algorithm. It would be interesting to see if the convergence properties of the algorithm in very hard cases can be further improved by exploring two straightforward extensions:

1) Adding to the cost function J terms that would reinforce a matching of the two-point correlations corresponding to the probabilities of observing mean probabilities of pairs $\langle S_i(t)S_j(t)\rangle$, $\langle S_i(t)I_j(t)\rangle$ and $\langle I_i(t)I_j(t)\rangle$ for $(ij) \in E$. In fact, these two-point correlations at equal times can be computed within the DMP approach: the basic relation is $\langle S_i(t)S_j(t)\rangle = P_S^{i\to j}(t) P_S^{j\to i}(t)$, where

$$P_S^{i\to j}(t) = P_S^i(0) \prod_{k\in\partial i\setminus j} \theta^{k\to i}(t) \quad (117)$$

has a meaning of the probability that the node i is in the state S at time t in an auxiliary cavity dynamics D_{ij}, in which the node j is fixed to the state S for all times; see [41] for more details. Once $\langle S_i(t)S_j(t)\rangle$ are computed, the other correlations are directly expressed as

$$\langle S_i(t)I_j(t)\rangle = P_S^i(t) - \langle S_i(t)S_j(t)\rangle,$$
$$\langle I_i(t)I_j(t)\rangle = 1 - P_S^i(t) - \langle I_i(t)S_j(t)\rangle. \quad (118)$$

2) Using the information contained in the second derivatives of the cost function J for a better control over the convergence. Indeed, the second derivatives can be computed in the same message-passing way as the gradient; it would, however, involve manipulations with the sixth-order tensors, as opposed to the fourth-order quantities used in the computation of the gradient (115) and (116). For example, if we denote $\frac{\partial^2\theta^{k\to i}(t)}{\partial\alpha_{rs}\partial\alpha_{uv}} \equiv p_{rs,uv}^{k\to i}(t)$ and $\frac{\partial^2\phi^{k\to i}(t)}{\partial\alpha_{rs}\partial\alpha_{uv}} \equiv q_{rs,uv}^{k\to i}(t)$, and since from (68) of the main text we have $m^i(t) = P_S^i(t-1) - P_S^i(t)$ for $t > 0$, where

$$P_S^i(t) = P_S^i(0) \prod_{k\in\partial i} \theta^{k\to i}(t), \quad (119)$$

it is sufficient to compute

$$\frac{\partial^2 P_S^i(t)}{\partial\alpha_{rs}\partial\alpha_{uv}} = P_S^i(0) \sum_{k\in\partial i} \Big[p_{rs,uv}^{k\to i}(t) \prod_{l\in\partial i\setminus k} \theta^{l\to i}(t) + p_{uv}^{k\to i}(t) \sum_{m\in i\setminus k} p_{rs}^{m\to i}(t) \prod_{l\in\partial i\setminus\{k,m\}} \theta^{l\to i}(t) \Big], \quad (120)$$

with $p_{rs,uv}^{k\to i}(t)$ following dynamic message-passing equations obtained as a derivative of (115) and (116) with respect to α_{uv}.

B. DMP algorithm for continuous dynamics

All the results in this Letter have been presented for the discrete-time model insofar. In this section, we briefly discuss how the same techniques can be easily extended to continuous case. The maximum likelihood estimator has been originally suggested for the continuous dynamics of the independent-cascade model [36]; our extension of the MLE for the case of missing information follows straightforwardly the main text, with discrete sums replaced by the integrals. For the DMP algorithm, we can use the continuous version of the DMP equations for the SI model, derived for the first time in [46]. An importance difference only concerns a choice of the objective function J: as in the case of missing information in time, it is more convenient to quantify the mismatch in terms of probabilities $P_S^i(t)$:

$$J'(t) = \sum_{t=0}^{n} J(t) = \sum_{t=0}^{n} \sum_{i\in\mathcal{O}} \frac{1}{2} \left[\widetilde{P}_S^i\left(\frac{t}{n}T\right) - P_S^i\left(\frac{t}{n}T\right) \right]^2, \quad (121)$$

where $n+1$ is a number of discretization steps in time which should be related to the statistics of activation times in M observed cascades, or to the set of observation times in the case of incomplete information in time, while $\widetilde{P}_S^i(t)$ and $P_S^i(t)$ are defined in the same way as for the incomplete information in time.

In the case of constant rates α_{ij}, we define the transmission function as $f_{ij}(t) = \alpha_{ij} e^{-\alpha_{ij} t}$. Then the functions $\theta^{i\to j}(t)$ are

computed as follows [46]:

$$\theta^{i\to j}(t) = 1 - \int_0^t d\tau\ f_{ij}(\tau)\left[1 - P_S^i(0)\prod_{k\in\partial i\setminus j}\theta^{k\to i}(t-\tau)\right]$$
$$= e^{-\alpha_{ij}t}$$
$$+ P_S^i(0)\alpha_{ij}e^{-\alpha_{ij}t}\int_0^t d\tau\ e^{\alpha_{ij}\tau}\left(\prod_{k\in\partial i\setminus j}\theta^{k\to i}(\tau)\right). \quad (122)$$

In order to compute the dynamic messages $\theta^{i\to j}(t)$, we can either integrate the expression (122) numerically, or transform the equation above into an ordinary differential equation by integrating the last term in (122) by parts:

$$\frac{d\theta^{i\to j}(t)}{dt} = -\alpha_{ij}\theta^{i\to j}(t) + \alpha_{ij}P_S^i(0)\prod_{k\in\partial i\setminus j}\theta^{k\to i}(t), \quad (123)$$

which can be solved numerically starting from initial conditions $\theta^{i\to j}(0) = 1$. The probabilities $P_S^i(t)$ are computed according to (119) in the continuous case as well.

The couplings are updated according to $\alpha_{rs}^{(t+\Delta t)} \leftarrow \alpha_{rs}^{(t)} - \epsilon\frac{\partial J'(t)}{\partial\alpha_{rs}}$ for $t\in[0,n]$ and fixed learning rate ϵ, where

$$-\frac{\partial J'(t)}{\partial\alpha_{rs}} = \sum_{i\in\mathcal{O}}\left[\widetilde{P}_S^i\left(\frac{t}{n}T\right) - P_S^i\left(\frac{t}{n}T\right)\right]\frac{\partial P_S^i\left(Tt/n\right)}{\partial\alpha_{rs}}. \quad (124)$$

The derivative of $P_S^i\left(Tt/n\right)$ reads:

$$\frac{\partial P_S^i\left(Tt/n\right)}{\partial\alpha_{rs}} = P_S^i(0)\sum_{k\in\partial i}p_{rs}^{k\to i}(Tt/n)\prod_{l\in\partial i\setminus k}\theta^{l\to i}(Tt/n), \quad (125)$$

where $p_{rs}^{k\to i}(t)$ obeys the following ordinary differential equation, obtained by taking a derivative of (123):

$$\frac{dp_{rs}^{k\to i}(t)}{dt} = -\alpha_{ki}p_{rs}^{k\to i}(t)$$
$$+\alpha_{ki}P_S^k(0)\sum_{m\in k\setminus i}p_{rs}^{m\to k}(t)\prod_{l\in\partial k\setminus\{i,m\}}\theta^{l\to k}(t)$$
$$+\mathbb{1}[k=r,i=s]\left[P_S^k(0)\prod_{l\in\partial k\setminus i}\theta^{l\to k}(t) - \theta^{k\to i}(t)\right]. \quad (126)$$

C. Reconstruction of transmission probabilities in the case of noisy information

In this section, we show that the DMP algorithm is naturally adapted for an efficient reconstruction in the case where the activation times are observed with some noise fluctuating around the true values of the activation time. The reason for that lies in the averaged empirical marginal probabilities used as an input for the DMP algorithm: while this averaging of the original data may represent a certain drawback since some detailed information on the cascades is lost, in the case of the observations corrupted by noise this procedure has a clear advantage because of the effective averaging over the fluctuations. As a simple test, we have perturbed a subset of the observed activation times (chosen among all activation times uniformly with probability $1/5$) $\{\tau_i^c\}_{i\in V}$ for $c = 1\dots M$ cascades with a noise $\{\Delta\tau_i^c\}_{i\in V}$ of randomly chosen sign, with absolute value distributed according to the Poisson distribution with mean $\mu = 1$. The results of a naive application of MLE and DMP algorithms are presented in the Fig. 4.

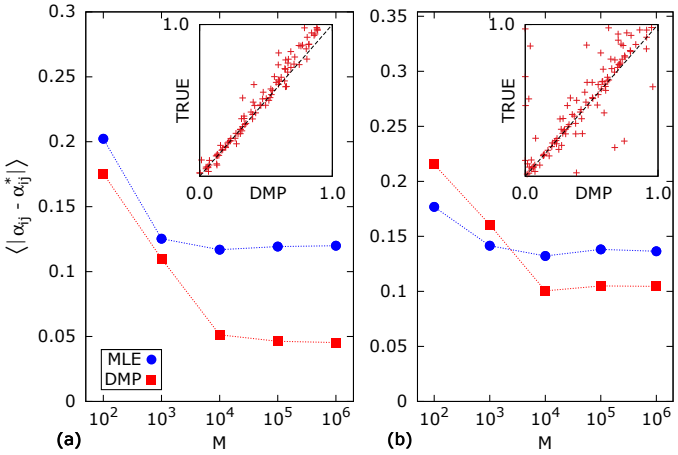

Fig. 4. (Color online) Comparison of mean error on the reconstructed couplings as a function of the number of fully observed cascades M for (a) a tree network with $N = 50$ and (b) a connected component of a power-law network with $N = 53$. Insets: Scatter plots of transmission probabilities reconstructed by DMP algorithm for $M = 10^6$ versus true couplings for (a) a tree and (b) a power-law network. The data corresponds to the perturbed cascades that has been generated from $\{\alpha_{ij}\}_{(ij)}$ uniformly distributed in the range $[0,1]$, and $T = 10$.

Acknowledgment

I am extremely grateful to M. Chertkov, M. Vuffray, A. Lokhov and H. Jang with whom I had the pleasure to work. I would like to thank also S. Mishra, A. Zlotnik, L. Road, A. Neukirch and G. Qu for countless discussions and fun. My work was partially supported by the National Science Foundation award No. 1128501 at New Mexico Consortium.

References

[1] M. I. Jordan, "Graphical models," *Statistical Science*, pp. 140–155, 2004.

[2] S. L. Lauritzen, *Graphical models*. Oxford University Press, 1996.

[3] M. J. Wainwright and M. I. Jordan, "Graphical models, exponential families, and variational inference," *Foundations and Trends® in Machine Learning*, vol. 1, no. 1-2, pp. 1–305, 2008.

[4] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D.-U. Hwang, "Complex networks: Structure and dynamics," *Physics reports*, vol. 424, no. 4, pp. 175–308, 2006.

[5] G. Bresler, "Efficiently learning ising models on arbitrary graphs," in *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing*. ACM, 2015, pp. 771–782.

[6] R. G. Gallager, *Information theory and reliable communication*. Wiley, 1968.

[7] S.-Y. Chung, J. Forney, G.D., T. Richardson, and R. Urbanke, "On the design of low-density parity-check codes within 0.0045 db of the shannon limit," *Communications Letters, IEEE*, vol. 5, no. 2, pp. 58–60, Feb 2001.

[8] T. Richardson, M. Shokrollahi, and R. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *Information Theory, IEEE Transactions on*, vol. 47, no. 2, pp. 619–637, Feb 2001.

[9] P. Oswald and A. Shokrollahi, "Capacity-achieving sequences for the erasure channel," *Information Theory, IEEE Transactions on*, vol. 48, no. 12, pp. 3017–3028, 2002.

[10] T. Richardson and R. Urbanke, "The capacity of low-density parity-check codes under message-passing decoding," *Information Theory, IEEE Transactions on*, vol. 47, no. 2, pp. 599–618, Feb 2001.

[11] S. Shamai and I. Sason, "Variations on the gallager bounds, connections, and applications," *Information Theory, IEEE Transactions on*, vol. 48, no. 12, pp. 3029–3051, Dec 2002.

[12] M. Chertkov and V. Y. Chernyak, "Loop series for discrete statistical models on graphs," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2006, no. 06, p. P06009, 2006. [Online]. Available: http://stacks.iop.org/1742-5468/2006/i=06/a=P06009

[13] B. D. McKay, "Subgraphs of random graphs with specified degrees," in *Proceedings of the International Congress of Mathematicians*, vol. 4, 2010, pp. 2489–2501.

[14] N. Macris and M. Vuffray, "The Bethe free energy allows to compute the conditional entropy of graphical code instances. A proof from the polymer expansion," *arXiv preprint arXiv:1310.1294*, 2013.

[15] ——, "Beyond the Bethe free energy of LDPC codes via polymer expansions," in *Information Theory Proceedings (ISIT), 2012 IEEE International Symposium on*, July 2012, pp. 2331–2335.

[16] M. Mézard and A. Montanari, *Information, physics, and computation*. Oxford University Press, 2009.

[17] J.-B. H. Urruty and C. Lemaréchal, *Fundamentals of convex analysis*. Springer, 2001.

[18] M. Vuffray, "The cavity method in coding theory," Ph.D. dissertation, IC, Lausanne, 2014, available at http://infoscience.epfl.ch/record/196951/files/EPFL_TH6088.pdf. [Online]. Available: http://infoscience.epfl.ch/record/196951/files/EPFL_TH6088.pdf

[19] D. Marbach, J. C. Costello, R. Küffner, N. M. Vega, R. J. Prill, D. M. Camacho, K. R. Allison, M. Kellis, J. J. Collins, and G. Stolovitzky, "Wisdom of crowds for robust gene network inference." *Nat. Methods*, vol. 9, no. 8, pp. 796–804, 2012.

[20] F. Morcos, A. Pagnani, B. Lunt, A. Bertolino, D. S. Marks, C. Sander, R. Zecchina, J. N. Onuchic, T. Hwa, and M. Weigt, "Direct-coupling analysis of residue coevolution captures native contacts across many protein families." *Proc. Natl. Acad. Sci. U.S.A.*, vol. 108, no. 49, pp. E1293–301, 2011.

[21] S. Cocco, S. Leibler, and R. Monasson, "Neuronal couplings between retinal ganglion cells inferred by efficient inverse statistical physics methods." *Proc. Natl. Acad. Sci. U.S.A.*, vol. 106, no. 33, pp. 14 058–62, 2009.

[22] N. Eagle, A. S. Pentland, and D. Lazer, "Inferring friendship network structure by using mobile phone data." *Proc. Natl. Acad. Sci. U.S.A.*, vol. 106, no. 36, pp. 15 274–8, 2009.

[23] S. Cocco and R. Monasson, "Adaptive Cluster Expansion for Inferring Boltzmann Machines with Noisy Data," *Phys. Rev. Lett.*, vol. 106, no. 9, p. 090601, 2011.

[24] G. Bresler, "Efficiently Learning Ising Models on Arbitrary Graphs," in *STOC*, 2015, pp. 771–782.

[25] M. Mézard and J. Sakellariou, "Exact mean-field inference in asymmetric kinetic Ising systems," *J. Stat. Mech.*, vol. 2011, no. 07, p. L07001, 2011.

[26] G. Bresler, D. Gamarnik, and D. Shah, in *Allerton*, Sep., pp. 1148–1155.

[27] H. W. Hethcote, "The Mathematics of Infectious Diseases," *SIAM Review*, vol. 42, no. 4, pp. 599–653, 2000.

[28] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D.-U. Hwang, "Complex networks: Structure and dynamics," *Phys. Rep.*, vol. 424, no. 4-5, pp. 175–308, 2006.

[29] E. M. Rogers, *Diffusion of innovations*. Simon & Schuster, New York, 2010.

[30] D. Strang and S. A. Soule, "Diffusion in Organizations and Social Movements: From Hybrid Corn to Poison Pills," *Ann. Rev. Soc.*, vol. 24, no. 1, pp. 265–290, 1998.

[31] D. Dhar, P. Shukla, and J. P. Sethna, "Zero-temperature hysteresis in the random-field Ising model on a Bethe lattice," *J. Phys. A*, vol. 30, no. 15, pp. 5259–5267, 1997.

[32] R. O'Dea, J. J. Crofts, and M. Kaiser, "Spreading dynamics on spatially constrained complex brain networks." *J. R. Soc. Interface*, vol. 10, no. 81, p. 20130016, 2013.

[33] Z. Shen, W.-X. Wang, Y. Fan, Z. Di, and Y.-C. Lai, "Reconstructing propagation networks with natural diversity and identifying hidden sources." *Nature communications*, vol. 5, p. 4323, 2014.

[34] X. Han, Z. Shen, W.-X. Wang, and Z. Di, "Robust Reconstruction of Complex Networks from Sparse Data," *Phys. Rev. Lett.*, vol. 114, no. 2, p. 028701, 2015.

[35] S. Myers and J. Leskovec, "On the Convexity of Latent Social Network Inference," in *NIPS*, 2010, pp. 1741–1749.

[36] M. Gomez-Rodriguez, D. Balduzzi, and B. Schölkopf, "Uncovering the Temporal Dynamics of Diffusion Networks," in *ICML*, 2011, pp. 561–568.

[37] M. Gomez-Rodriguez, J. Leskovec, and B. Schölkopf, "Structure and dynamics of information pathways in online media," in *WSDM*, 2013, p. 23.

[38] B. Dunn and Y. Roudi, "Learning and inference in a nonequilibrium Ising model with hidden nodes," *Phys. Rev. E*, vol. 87, no. 2, p. 022127, 2013.

[39] E. Sefer and C. Kingsford, "Convex Risk Minimization to Infer Networks from Probabilistic Diffusion Data at Multiple Scales," in *ICDE*, 2015.

[40] M. Farajtabar, M. Gomez-Rodriguez, N. Du, M. Zamani, H. Zha, and L. Song, "Back to the Past: Source Identification in Diffusion Networks from Partially Observed Cascades," in *AISTATS*, vol. 38, 2015.

[41] A. Y. Lokhov, M. Mézard, and L. Zdeborová, "Dynamic message-passing equations for models with unidirectional dynamics," *Phys. Rev. E*, vol. 91, no. 1, p. 012811, 2015.

[42] A. Y. Lokhov, M. Mézard, H. Ohta, and L. Zdeborová, "Inferring the origin of an epidemic with a dynamic message-passing algorithm," *Phys. Rev. E*, vol. 90, no. 1, p. 012801, 2014.

[43] See Supplemental Material at [URL will be inserted by publisher] for details on the implementation of algorithms and additional information.

[44] R. J. Williams and D. Zipser, "A learning algorithm for continually running fully recurrent neural networks," *Neural computation*, vol. 1, no. 2, pp. 270–280, 1989.

[45] W. W. Zachary, "An information flow model for conflict and fission in small groups," *J. Anthropol. Res.*, vol. 33, pp. 452–473, 1977.

[46] B. Karrer and M. E. J. Newman, "Message passing approach for general epidemic models," *Phys. Rev. E*, vol. 82, no. 1, p. 016101, 2010.

[47] H. Daneshmand, M. Gomez-Rodriguez, L. Song, and B. Schoelkopf, "Estimating Diffusion Network Structures: Recovery Conditions, Sample Complexity & Soft-thresholding Algorithm," in *ICML*, 2014, pp. 793–801.

[48] C. Chow and C. Liu, "Approximating discrete probability distributions with dependence trees," *Information Theory, IEEE Transactions on*, vol. 14, no. 3, pp. 462–467, 1968.

[49] J. Bento and A. Montanari, "Which graphical models are difficult to learn?" *arXiv preprint arXiv:0910.5761*, 2009.

[50] N. P. Santhanam and M. J. Wainwright, "Information-theoretic limits of selecting binary graphical models in high dimensions," *Information Theory, IEEE Transactions on*, vol. 58, no. 7, pp. 4117–4134, 2012.

[51] G. Bresler, E. Mossel, and A. Sly, "Reconstruction of markov random fields from samples: Some observations and algorithms," in *Approximation, Randomization and Combinatorial Optimization. Algorithms and Techniques*. Springer, 2008, pp. 343–356.

[52] A. Ray, S. Sanghavi, and S. Shakkottai, "Greedy learning of graphical models with small girth," in *Communication, Control, and Computing (Allerton), 2012 50th Annual Allerton Conference on*. IEEE, 2012, pp. 2024–2031.

[53] A. Anandkumar, V. Y. Tan, F. Huang, A. S. Willsky *et al.*, "High-dimensional structure estimation in ising models: Local separation criterion," *The Annals of Statistics*, vol. 40, no. 3, pp. 1346–1375, 2012.

[54] G. Bresler, D. Gamarnik, and D. Shah, "Structure learning of antiferromagnetic ising models," in *Advances in Neural Information Processing Systems*, 2014, pp. 2852–2860.

[55] P. Ravikumar, M. J. Wainwright, J. D. Lafferty *et al.*, "High-dimensional ising model selection using âĎŞ1-regularized logistic regression," *The Annals of Statistics*, vol. 38, no. 3, pp. 1287–1319, 2010.

[56] M. Ohzeki, "L 1-regularized boltzmann machine learning using majorizer minimization," *Journal of the Physical Society of Japan*, vol. 84, no. 5, p. 054801, 2015.