Editors' Suggestion

Inference in spreading processes with neural-network priors

Davide Ghio [1,*] Fabrizio Boncoraglio [2] and Lenka Zdeborová [2]

[1] *Information, Learning and Physics Laboratory, École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland*

[2] *Statistical Physics of Computation Laboratory, École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland*

(Received 11 September 2025; accepted 1 December 2025; published 7 January 2026)

Stochastic processes on graphs are a powerful tool for modeling complex dynamical systems such as epidemics. A recent line of work focused on the inference problem where one aims to estimate the state of every node at every time, starting from partial observation of a subset of nodes at a subset of times. In these works, the initial state of the process was assumed to be random independent and identically distributed over nodes. Such an assumption may not be realistic in practice, where one may have access to a set of covariate variables for every node that influences the initial state of the system. In this work, we will assume that the initial state of a node is an unknown function of such covariate variables. Given that functions can be represented by neural networks, we will study a model where the initial state is given by a simple neural network—notably the single-layer perceptron acting on the known nodewise covariate variables. Within a Bayesian framework, we study how such neural-network prior information enhances the recovery of initial states and spreading trajectories. We derive a hybrid belief propagation and approximate message passing (BP-AMP) algorithm that handles both the spreading dynamics and the information included in the node covariates, and we assess its performance against the estimators that use only the spreading information or use only the information from the covariate variables. We show that in some regimes, the model can exhibit first-order phase transitions when using a Rademacher distribution for the neural-network weights. These transitions create a statistical-to-computational gap where even the BP-AMP algorithm, despite the theoretical possibility of perfect recovery, fails to achieve it.

DOI: 10.1103/8mww-brdk

I. INTRODUCTION

Spreading dynamics on networks model many important systems, such as information diffusion in social networks [1], epidemic processes [2,3], and gene regulatory networks [4]. There are many studies in the literature that tackle the interplay between the network topology and the dynamical process that takes place on top of it [5].

In this work we study Bayesian inference for this class of models, aiming to recover some information about a realization of the dynamics from partial observations. In such Bayesian setting, this problem reduces to computing posterior expectations, where the spreading process itself defines the prior and observations form the likelihood. Statistical physics has yielded several inference algorithms for spreading models on random sparse graphs, including dynamic message passing (DMP) [6] and belief propagation (BP) [7]. Our work builds on the BP formulation to derive a new algorithm.

Spreading models often assume a separable prior over initial sources, ignoring node-specific information. However, node features are often available in practice and can enhance inference. To leverage such data, recent work in signal processing uses generative priors, often based on neural networks that map features to problem variables [8,9].

We introduce the Neural Sources Spreading (NSS) model, where the unknown initial state of the spreading process is given by the output of a neural network with unknown weights acting on known nodewise variables. The NSS model's posterior corresponds to a hybrid factor graph. It combines a sparse graph on which the spreading dynamics takes place—suited for BP on locally treelike structures [7,10]—with a dense graph for the neural-network prior, which is effectively handled by approximate message passing (AMP) in high dimensions [11,12]. Using the cavity method, we merge these components to derive a hybrid belief propagation and approximate message passing (BP-AMP) algorithm that handles both the sparse spreading interactions and the dense Neural-Network prior [13,14].

This paper is structured as follows. Section II defines the NSS model. Section III details the Bayesian inference framework and performance metrics. Section IV presents the cavity equations and our BP-AMP algorithm. We then analyze its performance in a setting showing evidence of Bayes optimality (Sec. V) and another exhibiting a statistical-to-computational gap (Sec. VI).

Related works

This study is situated within the field of spreading processes on graphs, which examines how these dynamics relate to the underlying graph structure, particularly in domains such as epidemiology [15,16]. Building on established compartmental models [17,18], we focus on the algorithmic challenges of inference, such as identifying the initial sources of an epidemic or recovering entire infection trajectories.

A large body of work in different scientific domains have studied the problem of inference of spreading processes on

*Contact author: davide.ghio98@gmail.com

graphs. Probably the most important example is the one of source detection, or equivalently of finding the patient zero of the epidemic, which has attracted many researchers after the seminal papers [19,20], where the authors, restricting to the susceptible-infected (SI) model, showed rigorous results for regular trees using the concept of rumor centrality as the maximum likelihood estimator. Later, in [21] the authors extended the study to general trees, and in [22] the susceptible-infected-recovered (SIR) model was considered, in both cases choosing as partial information a snapshot of the epidemic, i.e., the state of all the nodes at a particular time. At the same time in [23] the authors moved to consider localized observations, or sensors, as a source of information and described the optimal maximum-likelihood estimator for trees.

In [24], for the first time a message-passing approach was introduced to analyze epidemic models, enabling the computation of average properties on arbitrary random graph ensembles. Later, two different works were able to design a message-passing algorithm operating on single graphs: in [6] the authors introduced a "Dynamical Message Passing" (DMP) algorithm, which is exact on trees but gives results for a fixed initial condition, while in [25] the "Belief Propagation" (BP) equations for this problem were derived for the first time, and were used to infer the origin of epidemics with multiple patients zero and various types of partial information. These techniques were later used for other related problems, as the reconstruction of the model's parameters [26] or epidemic mitigation studies [27].

The methodology we employ connects to the idea of merging graphical models, an approach that was formulated in [13,28] to extend single-layer AMP to multiple layers. Closer to the present setting, in [29,30], for the first time a message-passing algorithm was devised in a model combining sparse and dense graphical models, while [14] derived a BP-AMP algorithm for community detection with a neural-network prior on node attributes.

Finally, we note a recent line of work conjecturing the existence of replica symmetry breaking in spreading processes at very low source densities [31,32]. This paper, in contrast, focuses on scenarios with higher source densities where this phenomenology is not observed, leaving the low-density regime with a neural-network prior for future study.

II. THE NEURAL SOURCES SPREADING MODEL

A. Spreading models on graphs

We consider processes on a graph $G(V, E)$ with N nodes. Each node of the graph is assigned a variable x_i^t at discrete times $t \in \{0, 1, \ldots, T\}$, taking values among a finite number p of possible states. We focus on unidirectional dynamics, where a node's transition times between the possible states of the nodes $\{\mathbf{t_i}\}_{i=1}^N = \{t_i^1, \ldots, t_i^{p-1}\}_{i=1}^N$, which we define as the last time in which a node is in a state before it transitions, fully specify its evolution. This static representation reduces the system's variables from $O(N \times T)$ to $O(N)$. This restriction can be relaxed to models where nodes revisit states (e.g., Susceptible-Infected-Susceptible (SIS)) by treating backward steps as new state transitions [2,33].

Assuming local spreading dynamics, where transitions depend only on nearest-neighbor states, the transition time probability factorizes as

$$P\left(\{\mathbf{t_i}\}_{i=1}^N \middle| \mathbf{\Theta}\right) = \frac{1}{Z_{\text{spread}}} \prod_{i=1}^{N} \Psi_i(\mathbf{t_i}, \{\mathbf{t_j}\}_{j\in\partial_i}, \mathbf{\Theta}), \tag{1}$$

where Z_{spread} is the normalization, $\mathbf{\Theta}$ are model parameters, and ∂_i is the set of neighbors of node i. Importantly, the factor $\Psi_i(\mathbf{t_i}, \{\mathbf{t_j}\}_{j\in\partial_i}, \mathbf{\Theta})$ is a node factor that depends on its transition time and those of its nearest neighbors.

We will illustrate the method on two examples of the dynamical model: the SI model and the deterministic susceptible-infected-recovered (dSIR) model [33].

(1) In the SI model, each node is either susceptible (S) or infectious (I), so the dynamics is described by a single transition time t_i for each node i. We say that node i is a *source* of the spreading if it is infectious at time $t = 0$, and in this case we have $t_i = -1$. At time t, a susceptible node i has a probability $\lambda_{ji}^t \in [0, 1]$ to get infected from an infectious node $j \in \partial_i$. In mathematical terms, the probability for node i of being infected at time s is given by

$$p_i^{\text{inf}}\left(s, \{t_k\}_{k\in\partial_i}\right) = 1 - \prod_{k\in\partial_i} (1 - \lambda_{ki} \mathbb{I}[s > t_k]). \tag{2}$$

The total factor for this node is then

$$\Psi_i\left(t_i, \{t_k\}_{k\in\partial_i} \middle| x_i^0\right) = \begin{cases} p_i^{\text{inf}}\left(t_i, \{t_k\}_{k\in\partial_i}\right) \prod_{s=0}^{t_i-1} \left(1 - p_i^{\text{inf}}\left(s, \{t_k\}_{k\in\partial_i}\right)\right) & \text{if } x_i^0 = S, \quad t_i < T \\ \prod_{s=0}^{T-1} \left(1 - p_i^{\text{inf}}\left(s, \{t_k\}_{k\in\partial_i}\right)\right) & \text{if } x_i^0 = S, \quad t_i = T \\ \delta_{t_i,-1} & \text{if } x_i^0 = I, \end{cases} \tag{3}$$

where we assign $t_i = T$ to nodes that are still susceptible at $t = T - 1$, the last simulation time.

(2) In the dSIR model, S-to-I transitions are still stochastic, and follow the same rule as the SI model. However, an additional recovery process takes place and it is deterministic: a node remains infectious for a fixed time Δ_i before recovering permanently. The resulting factor for node i can still be written as in Eq. (3), but with the modified infection probability

$$p_i^{\text{inf}}\left(s, \{t_k\}_{k\in\partial_i}\right) = 1 - \prod_{k\in\partial_i} (1 - \lambda_{ki} \mathbb{I}[t_k < s \leqslant t_k + \Delta_k]). \tag{4}$$

We can recover the SI model by fixing $\Delta_i > T \; \forall i$, such that all nodes remain infectious until the end.

Throughout, we assume the form of the spreading kernel in Eq. (1) is known, as well as all parameters $\boldsymbol{\Theta}$ of the spreading model. In the SI model, these include the graph G and all the associated infectivity parameters $\lambda_{ij}\ \forall (i,j) \in E$, while for the dSIR model we also have the recovery delays $\Delta_i\ \forall i \in V$. Moreover, we will focus on studying inference on graphs generated through random sparse ensembles, such as Erdös-Rényi (ER) and Random Regular (RR) graphs.

B. Neural-network priors on the spreading sources

Spreading models often assume that the nodes that originate the spreading, i.e., the sources, are single [6,19,23] or multiple [7,34] uniformly random nodes. Realistically, the source prior is often nontrivial and can be informed by known nodewise covariates $F \in \mathbb{R}^{N\times M}$, namely, each node $i \in V$ has M features, contained in the vector $\mathbf{F_i} \in \mathbb{R}^M$, representing some contextual information about each node. For instance, in epidemiology, covariates like age or mobility can determine a node's propensity to act as an initial source. Such features can be incorporated to improve inference [35]. This implies the initial state is a function of its covariates, $x_i^0 = f(\mathbf{F_i})$. In the SI and dSIR models described before, this variable is binary: node i can either be a source of the spreading, and we conventionally assign $x_i^0 = -1$, or it is susceptible at the start, and in this case $x_i^0 = +1$.

Since generic functions can be parametrized using a multilayer neural network, it is reasonable to consider the covariates $\mathbf{F_i}$ to be an input and the initial states $\mathbf{x_i^0}$ to be an output of a multilayer fully connected neural network:

$$x_i^0 = \varphi^{(L)}\big(W^{(L)}\varphi^{(L-1)}\big(W^{(L-1)}\dots\varphi^{(1)}\big(W^{(1)}\mathbf{F_i}\big)\dots\big)\big), \quad (5)$$

where $\varphi^{(l)}(\cdot)$ are componentwise nonlinearities and $\{W^{(l)}\}_{l=1}^L$ are the network weights. We call this the NSS model.

In order to obtain a model where the information contained in the prior can be quantified precisely, we will consider in what follows a single-layer network with random weights $\mathbf{u} \in \mathbb{R}^M$, following a known, componentwise prior $u_a \sim P_u$, $\forall\, a \in \{1,\dots,M\}$. The initial state is then sampled according to

$$x_i^0 \sim P^{\text{out}}\big(x_i^0|\mathbf{F_i},\mathbf{u}\big) \equiv \delta\Big[x_i^0 - \varphi\Big(\sum_a F_{ia}u_a\Big)\Big]. \quad (6)$$

Here, $\delta(\cdot)$ is the Dirac delta. For modeling purposes we also choose $F \in \mathbb{R}^{N\times M}$ as a matrix of i.i.d. Gaussian components $F_{ia} \sim \mathcal{N}(0, 1/M)$. We will consider the two cases of Gaussian and Rademacher random weights $\mathbf{u}$.

We set $\varphi(x) = \text{sgn}(x-\kappa)$, yielding the perceptron model [36–38], where the threshold κ controls the source density. We call this specific version the Perceptron Sources Spreading (PSS) model.

In order to be in an analytically tractable setting, we study the limit where $N, M \to \infty$ with a finite constant ratio $\alpha = N/M = \mathcal{O}(1)$. In this regime, α acts as a signal-to-noise ratio controlling correlations between initial states. For large α (small M), the x_i^0 are strongly correlated via $\mathbf{u}$. As $\alpha \to 0$, the terms $\sum_{a=1}^M F_{ia}u_a$ become independent Gaussians, and we recover a uniform prior $P(x_i^0 = I) = \delta$ for all i, with $\delta = \frac{1}{2}[1 + \text{erf}(\frac{\kappa}{\sqrt{2}})]$. In the intermediate regime, from previous results on the perceptron [12], we expect two different phenomenologies when changing α, depending on the prior chosen for the weights: for Gaussian weights, the induced landscape is approximately smooth and convex, leading to a continuous (second-order) transition; while for Rademacher weights, the resulting discrete and highly frustrated structure introduces competing minima, giving rise to a discontinuous (first-order) phase transition.

III. BAYESIAN INFERENCE FRAMEWORK

In this section, we define in detail the inference problem. We assume we are given the covariates $F \in \mathbb{R}^{N\times M}$, the graph structure G, and the edgewise transmission probabilities λ_{ij}. We also know the constant κ from Eq. (6), and all Δ_i for the dSIR model. We are not given the vector $\mathbf{u}$, and consequently the sources are also unknown, as well as the transition times $\mathbf{t}_i$. The task is to recover the initial state $\mathbf{x}^0$ and the transition times $\mathbf{t}_i$ based on additional partial observation of the state of each node $\{\mathcal{O}_i\}_{i=1}^N$. We assume the availability of nodewise observations $\boldsymbol{\mathcal{O}} = \{\mathcal{O}_i\}_{i=1}^N$,

$$P(\boldsymbol{\mathcal{O}}|\{\mathbf{t_i}\},\mathbf{x^0},\boldsymbol{\Theta}) = \prod_{i=1}^N P\big(\mathcal{O}_i|\mathbf{t_i},x_i^0,\boldsymbol{\Theta}\big), \quad (7)$$

where $\{\mathbf{t_i}\}$ is the set of all transition times. This implies that the observation $\mathcal{O}_i$ is conditionally independent of other nodes' transition times given $\mathbf{t_i}$. In the following, we will focus on two classes of observations: the first are *sensors* [23,39], in which we choose randomly a fraction ρ of the nodes, and we observe their full time trajectory, such that an observation on node i acts as a delta function on the transition time t_i. The second framework, known as *snapshots* [6,25], is when the entire system state $\mathbf{x^{t=T_{obs}}}$ is observed at a single time T_{obs}, and thus an observation on node i does not fix t_i, but gives either a lower bound $t_i \geqslant T_{\text{obs}}$ or an upper bound $t_i < T_{\text{obs}}$.

Using Bayes' rule, the posterior distribution of the model becomes

$$P(\{\mathbf{t_i}\},\mathbf{x^0},\mathbf{u}|\boldsymbol{\mathcal{O}},F,\boldsymbol{\Theta}) = \frac{1}{P(\boldsymbol{\mathcal{O}}|F,\boldsymbol{\Theta})}P(\boldsymbol{\mathcal{O}}|\{\mathbf{t_i}\},\mathbf{x^0},\mathbf{u},F,\boldsymbol{\Theta})P(\{\mathbf{t_i}\},\mathbf{x^0},\mathbf{u}|F,\boldsymbol{\Theta}) \quad (8)$$

$$= \frac{1}{P(\boldsymbol{\mathcal{O}}|F,\boldsymbol{\Theta})}P(\boldsymbol{\mathcal{O}}|\{\mathbf{t_i}\},\mathbf{x^0},\boldsymbol{\Theta})P(\{\mathbf{t_i}\}|\mathbf{x^0},\boldsymbol{\Theta})P(\mathbf{x^0}|\mathbf{u},F)P(\mathbf{u}) \quad (9)$$

$$= \frac{1}{Z(\boldsymbol{\mathcal{O}},F,\boldsymbol{\Theta})}\prod_{i=1}^N \widetilde{\Psi}_i\big(\mathbf{t_i},\{\mathbf{t_j}\}_{j\in\partial_i},x_i^0,\mathcal{O}_i,\boldsymbol{\Theta}\big)\Psi_i^{\text{out}}\big(x_i^0,\mathbf{u},\mathbf{F_i}\big)\prod_{a=1}^M \psi_a(u_a), \quad (10)$$

where $\psi_a(u_a) = P^u(u_a)$ is the prior distribution of the weights, $\Psi_i^{\text{out}}(x_i^0, \mathbf{u}, \mathbf{F_i}) = P^{\text{out}}(x_i^0|\mathbf{F_i}, \mathbf{u})$ the output distribution of the initial epidemic state of the nodes given the external context, and $\widetilde{\Psi}_i(\mathbf{t_i}, \{\mathbf{t_j}\}_{j\in\partial_i}, x_i^0, \mathcal{O}_i, \boldsymbol{\Theta}) \equiv \Psi_i(\mathbf{t_i}, \{\mathbf{t_j}\}_{j\in\partial_i}, \boldsymbol{\Theta})P(\mathcal{O}_i|\mathbf{t_i}, x_i^0, \boldsymbol{\Theta})$, which contains the information regarding the epidemic and the observations provided. Finally,

$$Z(\boldsymbol{\mathcal{O}}, F, \boldsymbol{\Theta}) \equiv \int \left(\prod_{a=1}^{M} \mathrm{d}u_a \psi_a(u_a)\right) \times \sum_{\{\mathbf{t_i}, x_i^0\}_{i=1}^N} \prod_{i=1}^{N} \widetilde{\Psi}_i\left(\mathbf{t_i}, \{\mathbf{t_j}\}_{j\in\partial_i}, x_i^0, \mathcal{O}_i, \boldsymbol{\Theta}\right) \times \Psi_i^{\text{out}}\left(x_i^0, \mathbf{u}, \mathbf{F_i}\right) \tag{11}$$

is the partition function, and we define

$$\Phi(\boldsymbol{\mathcal{O}}, F, \boldsymbol{\Theta}) = \frac{1}{N} \log Z(\boldsymbol{\mathcal{O}}, F, \boldsymbol{\Theta}) \tag{12}$$

the associated free entropy.

Sources retrieval and optimal inference

A Bayesian approach unifies various inference tasks, as they all reduce to estimating marginals of initial states $\{x_i^0\}$ or transition times $\{\mathbf{t_i}\}$. For concreteness, we focus on retrieving the spreading sources. An analysis of retrieving the full trajectory, which shows similar phenomenology, is in Appendix C 3.

To quantify source recovery performance, since the initial states x_i^0 are binary (susceptible or source), a natural metric is the overlap between the ground-truth state $\mathbf{x^{*,0}}$ and an estimator $\hat{\mathbf{x}}^{\mathbf{0}}$. The overlap is defined as

$$\mathrm{O}(\mathbf{x}, \mathbf{y}) \equiv \frac{1}{N} \sum_{i=1}^{N} \delta_{x_i, y_i}, \tag{13}$$

where $\mathbf{x}, \mathbf{y}$ are two generic vectors and δ_{x_i,y_i} is the Kronecker delta.

The Bayes-optimal strategy uses both observations and features to compute the posterior $P(\mathbf{x^0}|\boldsymbol{\mathcal{O}}, F)$ and leads to an estimator $\hat{\mathbf{x}}^{\mathbf{0}}$ that maximizes the mean overlap with the true state:

$$\mathrm{MO}(\hat{\mathbf{x}}^{\mathbf{0}}) \equiv \mathbb{E}_{\mathbf{x^0}|\boldsymbol{\mathcal{O}},F}[\mathrm{O}(\hat{\mathbf{x}}^{\mathbf{0}}, \mathbf{x^0})] = \frac{1}{N} \sum_{\mathbf{x^0}} P(\mathbf{x^0}|\boldsymbol{\mathcal{O}}, F) \sum_{i=1}^{N} \delta_{\hat{x}_i^0, x_i^0}. \tag{14}$$

It is known [11] that this quantity is maximized by the estimator

$$\hat{x}_i^{0,\mathrm{MMO}} = \underset{x_i^0}{\mathrm{argmax}}\, \mu\left(x_i^0|\boldsymbol{\mathcal{O}}, F\right) \quad \forall i \in [1:N], \tag{15}$$

where $\mu(x_i^0|\boldsymbol{\mathcal{O}}, F)$ is the posterior marginal for node i. This is the Maximum Mean Overlap (MMO) estimator. We analyze its performance via the overlap $\mathrm{O}_{t=0} \equiv \mathrm{O}(\hat{\mathbf{x}}^{\mathbf{0,MMO}}, \mathbf{x^{*,0}})$ and mean overlap $\mathrm{MO}_{t=0} \equiv \mathrm{MO}(\hat{\mathbf{x}}^{\mathbf{0,MMO}})$.

In a Bayes-optimal setting, the Nishimori conditions have to hold [11,40]. Since we will resort to approximations, due to finite-size effects and the replica symmetric (RS) assumption, our algorithm is expected to be "asymptotically optimal" in the number of variables, and verifying the validity of the Nishimori conditions is a useful self-consistency check. In a Bayes-optimal setting, indeed, the estimated prior and likelihood match the true distributions, making samples from the posterior statistically indistinguishable from the ground truth. This implies that the expected overlap $\mathrm{O}_{t=0}$ and the expected mean overlap $\mathrm{MO}_{t=0}$ must coincide in expectation. The expected overlap over the disorder is

$$\mathbb{E}_{\mathbf{x^{*,0}},\boldsymbol{\mathcal{O}},F}[\mathrm{O}(\hat{\mathbf{x}}^{\mathbf{0,MMO}}(\boldsymbol{\mathcal{O}}, F), \mathbf{x^{*,0}})] = \mathbb{E}_F\left[\sum_{\mathbf{x^{*,0}},\boldsymbol{\mathcal{O}}} \mathrm{O}(\hat{\mathbf{x}}^{\mathbf{0,MMO}}(\boldsymbol{\mathcal{O}}, F), \mathbf{x^{*,0}}) \times P^*(\mathbf{x^{*,0}})P^*(\boldsymbol{\mathcal{O}}|\mathbf{x^{*,0}})\right], \tag{16}$$

where P^* denotes the ground-truth distributions. Similarly, the disorder-averaged mean overlap is

$$\mathbb{E}_{\boldsymbol{\mathcal{O}},F}\left\{\mathrm{MO}[\hat{\mathbf{x}}^{\mathbf{0,MMO}}(\boldsymbol{\mathcal{O}}, F)]\right\} = \mathbb{E}_{\boldsymbol{\mathcal{O}},F}\left[\sum_{\mathbf{x^0}} \mathrm{O}(\hat{\mathbf{x}}^{\mathbf{0,MMO}}(\boldsymbol{\mathcal{O}}, \mathrm{F}), \mathbf{x^0})\mathrm{P}(\mathbf{x^0}|\boldsymbol{\mathcal{O}}, \mathrm{F})\right] \tag{17}$$

$$= \mathbb{E}_F\left[\sum_{\mathbf{x^0},\boldsymbol{\mathcal{O}}} (\hat{\mathbf{x}}^{\mathbf{0,MMO}}(\boldsymbol{\mathcal{O}}, F), \mathbf{x^0})P(\boldsymbol{\mathcal{O}}|\mathbf{x^0})P(\mathbf{x^0})\right]. \tag{18}$$

Bayes optimality implies $P(\cdot) = P^*(\cdot)$, so Eq. (16) and Eq. (18) coincide. Substituting the MMO estimator yields

$$\mathbb{E}[\mathrm{O}_{t=0}] = \frac{1}{N} \sum_{i=1}^{N} \mathbb{E}\left[\mathbb{I}\left[\arg\max_{x_i^0} \mu\left(x_i^0|\boldsymbol{\mathcal{O}}, F\right) = x_i^{*,0}\right]\right] \tag{19}$$

and

$$\mathbb{E}[\mathrm{MO}_{t=0}] = \frac{1}{N} \sum_{i=1}^{N} \mathbb{E}\left[\max_{x_i^0} \mu\left(x_i^0|\boldsymbol{\mathcal{O}}, F\right)\right], \tag{20}$$

where $\mathbb{I}(\cdot)$ is the indicator function. In our numerical experiments, we utilize the equivalence of the empirical estimates in Eqs. (19) and (20) to assess the consistency of the approximations made in the algorithm with the expected behavior of the Bayes-optimal estimator.

IV. THE ALGORITHM

In this section we present the inference algorithm we are going to use to estimate the posterior marginals of the problem. This is derived using the cavity method from statistical physics [10] and combines AMP for Generalized Linear Models [11,12] with BP for spreading models [25,33]. This approach of merging dense and sparse graphical models has been successfully applied in related contexts [14,28,29,41].

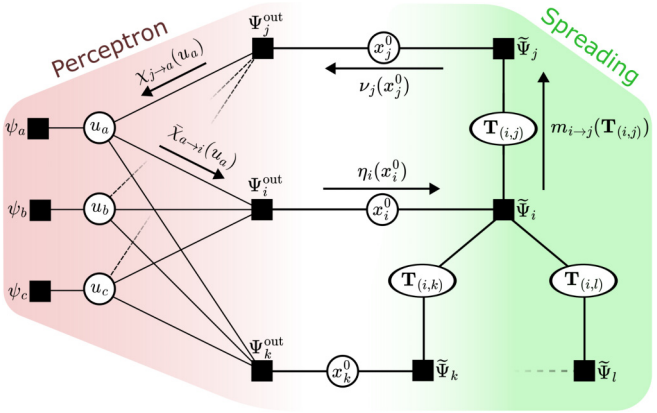

FIG. 1. Factor Graph. Representation of the posterior distribution for the Neural Sources Spreading model with a one-layer perceptron prior as a factor graph, with the associated belief propagation messages.

The factor graph for the posterior measure in Eq. (10) is shown in Fig. 1. Standard cavity arguments [10] yield the BP equations:

$$
\begin{aligned}
m_{i\to j}(t_i,t_j) &= \frac{1}{Z_{i\to j}}\sum_{x_i^0}\eta_i\big(x_i^0\big)\sum_{\mathbf{t_{k\in\partial i\setminus j}}}\tilde{\Psi}_i\big(t_i,\mathbf{t_{k\in\partial i}},\mathcal{O}_i,x_i^0\big)\\
&\quad\times\prod_{k\in\partial_i\setminus j}m_{k\to i}(t_k,t_i),\\
\nu_i\big(x_i^0\big) &= \frac{1}{Z_i^{\nu}}\sum_{t_i,\mathbf{t_{k\in\partial i}}}\tilde{\Psi}_i\big(t_i,\mathbf{t_{k\in\partial i}},\mathcal{O}_i,x_i^0\big)\prod_{k\in\partial i}m_{k\to i}(t_k,t_i),\\
\eta_i\big(x_i^0\big) &= \frac{1}{Z_i^{\eta}}\prod_a\left(\int \mathrm{d}u_a\,\bar{\chi}_{a\to i}(u_a)\right)\Psi_i^{\text{out}}\big(x_i^0,\mathbf{u},\mathbf{F_i}\big),\\
\chi_{i\to a}(u_a) &= \frac{1}{Z_{i\to a}}\sum_{x_i^0}\nu_i\big(x_i^0\big)\int\left(\prod_{b\neq a}\mathrm{d}u_b\,\bar{\chi}_{b\to i}(u_b)\right)\\
&\quad\times\Psi_i^{\text{out}}\big(x_i^0,\mathbf{u},\mathbf{F_i}\big),\\
\bar{\chi}_{a\to i}(u_a) &= \frac{1}{\bar{Z}_{a\to i}}\psi_a(u_a)\prod_{j\neq i}\chi_{j\to a}(u_a).
\end{aligned}
\tag{21}
$$

These equations allow one to compute the probability marginals exactly when the factor graph associated to the posterior is acyclic [42]. In our setting, we will consider spreading graphs that are locally treelike, i.e., with loops of typical length scaling as $\log N$, for which BP is expected to estimate the marginals with an error that goes to zero as N goes to ∞, if the RS assumption holds. We will refer to this property as the asymptotic optimality of the algorithm. Recently, some evidence was shown [31,32] towards the existence of a replica symmetry broken phase in spreading models for very low values of the fraction of sources, where these assumptions break down and the BP equations stop converging. Here, we focus on cases where the density of sources is high enough that these issues are not observed, thus remaining in the regime where the RS assumption is valid.

Equations (21) are generally intractable due to high-dimensional integrals. They simplify by applying the central limit theorem and projecting messages onto their first two moments—a standard procedure [11]. These simplifications yield key denoising functions, for which we postpone the detailed derivation in Appendix A.

The output denoising function is

$$
\begin{aligned}
g_o(\omega_i,\nu_i,V_i) &= \frac{\int \mathrm{d}z\sum_x \nu_i(x)P_{\text{out}}(x|z)(z-\omega_i)e^{-[(z-\omega_i)^2/2V_i]}}{V_i\int \mathrm{d}z\sum_x \nu_i(x)P_{\text{out}}(x|z)e^{-[(z-\omega_i)^2/2V_i]}}\\
&= \frac{2(2\nu_i-1)\exp\left(-\frac{(\omega_i-\kappa)^2}{2V_i}\right)}{\sqrt{2\pi V_i}\left[1+(2\nu_i-1)\,\mathrm{erf}\left(\frac{\omega_i-\kappa}{\sqrt{2V_i}}\right)\right]},
\end{aligned}
\tag{22}
$$

where the last step uses $P_{\text{out}}(x|z)=\delta[x-\text{sgn}(z-\kappa)]$. The input denoising functions are

$$
\begin{aligned}
f_a(A,B) &= \frac{\int \mathrm{d}u\,P^u(u)u\,\exp\left[-\frac{A}{2}u^2+Bu\right]}{\int \mathrm{d}u\,P^u(u)\exp\left[-\frac{A}{2}u^2+Bu\right]},\\
f_v(A,B) &= \frac{\partial}{\partial B}f_a(A,B).
\end{aligned}
\tag{23}
$$

For a Gaussian prior, $f_a(A,B)=\frac{B}{A+1}$ and $f_v(A,B)=\frac{1}{A+1}$. For a Rademacher prior, $f_a(A,B)=\tanh(B)$ and $f_v(A,B)=1-\tanh^2(B)$. The final algorithm is presented in Algorithm 1, and it allows one to directly get the cavity estimation of the marginals for the variables of the problem. For the sake of computing the overlaps defined in Sec. III A, all we need are the values at convergence of the BP-AMP messages $\eta_i(x_i^0)$ and $\nu_i(x_i^0)$ for every node. Then, the estimation of the marginal $\mu(x_i^0|\mathcal{O},F)$ will be given by

$$
\chi_i\big(x_i^0\big) \equiv \frac{\eta_i\big(x_i^0\big)\nu_i\big(x_i^0\big)}{\sum_{x_i^0}\eta_i\big(x_i^0\big)\nu_i\big(x_i^0\big)}.
\tag{24}
$$

In turn, we can use this quantity in Eqs. (19) and (20) to get an estimation of the average overlap and mean overlap.

Free entropy

We have defined the free energy of the problem in Eq. (12) as the logarithm of the partition function, which in general is a quantity that is too expensive to compute by brute force. The algorithm just defined allows us to estimate the large size limit of the free entropy directly from the values of the BP-AMP messages at convergence. A standard replica-symmetric cavity computation (see Appendix B) gives the final expression:

$$
\begin{aligned}
\phi_{\text{RS}} &= \frac{1}{N}\sum_{i=1}^{N}\log Z_i^{v} - \frac{1}{N}\sum_{(i,j)\in E}\phi_{(i,j)}^{\text{spread}}\\
&\quad+\sum_{a=1}^{M}\log\int \mathrm{d}u_a P^u(u_a)\exp\left[-\frac{A}{2}u_a^2+B_a u_a\right]\\
&\quad+\frac{1}{N}\sum_{i=1}^{N}\phi_i^{\text{out}}+\sum_{a=1}^{M}\left[\frac{A}{2}\big(a_a^2+v_a\big)-B_a a_a\right]\\
&\quad+\sum_{i=1}^{N}\frac{\left(\omega_i-\sum_{a=1}^{M}F_{ia}a_a\right)^2}{2V_i},
\end{aligned}
\tag{25}
$$

where

$$\phi^{\text{spread}}_{(i,j)} = \log \sum_{t_i,t_j} m_{i\to j}(t_i,t_j)\, m_{j\to i}(t_j,t_i) \tag{26}$$

and

$$\phi^{\text{out}}_i = \log\left(\sum_{x^0_i} \nu_i(x^0_i) \int \left(\prod_b \mathrm{d}u_b\right) \Psi^{\text{out}}_i(x^0_i, \mathbf{u}, \mathbf{F_i}) \times \prod_b \bar{\chi}_{b\to i}(u_b)\right) \tag{27}$$

$$= \log\left(\sum_{x^0_i} \nu_i(x^0_i)\eta_i(x^0_i)\right) - \log Z^{\eta}_i. \tag{28}$$

Algorithm 1 typically has a single fixed point corresponding to the free entropy maximum. However, first-order phase transitions can lead to multiple fixed points. Comparing their free entropy values, and selecting the fixed point with larger free entropy, characterizes the Bayes-optimal estimator, a point we revisit at the end of Sec. VI.

V. GAUSSIAN WEIGHTS PERCEPTRON PRIOR

We analyze Algorithm 1's performance for source retrieval with Gaussian weights [$P^u(u_a) = e^{-u_a^2/2}/\sqrt{2\pi}$]. Our experimental setup is as follows:

(1) *Graph*. We use RR graphs with degree $d = 3$ and homogeneous weights $\lambda^t_{ij} = \lambda$. On these locally treelike graphs, BP is expected to be asymptotically optimal in the absence of first-order phase transitions [10,43] and of replica symmetry breaking [31,32]. Appendix C 5 shows results for other ensembles, suggesting marginal dependence on network topology.

(2) *Observations*. We consider observations of two types:

(a) *Sensors* [23,39]. The full time trajectory of a fraction ρ of random nodes is observed.

(b) *Snapshot* [6,25]. The entire system state $\mathbf{x}^{\mathbf{t=T_{obs}}}$ is observed at a single time T_{obs}.

(3) *Spreading dynamics*. The dynamics follow the dSIR model (Sec. II) with a fixed recovery time $\Delta_i = \Delta$. We compare the case $\Delta = 1$, in which nodes remain infectious only for a time step, with the standard SI model. In the former, we run the dynamics until there is no more infectious individual in the network, while in the latter we wait until no susceptible individual remains. We define T as the first time step in which either of the two happens.

ALGORITHM 1. BP-AMP algorithm for the PSS model.

Input : Features $\mathbf{F}$, Graph $G(N,M)$, Observations $\mathcal{O}$, Functions g_o, f_a, f_v
Initialisation: $\mathbf{g_o^{(0)}} = \mathbf{0}$, $\mathbf{a^{(0)}}$, $\mathbf{v^{(0)}}$, $\boldsymbol{\nu}^{(0)}$, $\{\mathbf{m^{(0)}_{i\to j}}\}_{(i,j)\in E}$
Output : $\hat{\mathbf{x}}^0$
Repeat until convergence
  $V^{(t+1)} = \frac{1}{M}\sum_a v_a^{(t)}$;
  for $i = 1,\dots,N$ **do**
    $\nu_i^{(t+1)}(x_i^0) = \sum_{t_i,\mathbf{t_{k\in\partial i}}} \widetilde{\psi}_i(t_i, \mathbf{t_{k\in\partial i}}, \mathcal{O}_i, x_i^0) \prod_{k\in\partial i} m^{(t)}_{k\to i}(t_k,t_i)$;
    `Normalize` ν_i;
  end
  for $i = 1,\dots,N$ **do**
    $\omega_i^{(t+1)} = \sum_a F_{ia} a_a^{(t)} - V^{(t+1)} g^{(t)}_{o,i}$;
    $g^{(t+1)}_{o,i} = g_o(\omega_i^{(t+1)}, \nu_i^{(t+1)}, V^{(t+1)})$;
    $\eta_i^{(t+1)}(x_i^0) = \int \mathrm{d}z P_{\text{out}}(x_i^0 \mid z) e^{-\frac{(z-\omega_i^{(t+1)})^2}{2V^{(t+1)}}}$;
    `Normalize` $\eta_i^{(t+1)}$;
    $\chi_i^+ = \frac{\eta_i(+1)\nu_i(+1)}{\eta_i(+1)\nu_i(+1)+\eta_i(-1)\nu_i(-1)}$;
    $\hat{x}_i^{0,(t+1)} = 2\chi_i^+ - 1$;
  end
  $A^{(t+1)} = \frac{1}{M}\sum_i \left(g^{(t+1)}_{0,i}\right)^2$;
  for $a = 1,\dots,M$ **do**
    $B_a^{(t+1)} = \sum_i F_{ai} g^{(t+1)}_{o,i} + a_a^{(t)} A^{(t+1)}$;
    $a_a^{(t+1)} = f_a(A^{(t+1)}, B_a^{(t+1)})$;
    $v_a^{(t+1)} = f_v(A^{(t+1)}, B_a^{(t+1)})$;
  end
  for $(i,j) \in E$ **do**
    $m^{(t+1)}_{i\to j}(t_i,t_j) = \sum_{x_i^0} \eta_i^{(t+1)}(x_i^0) \sum_{t_i,\mathbf{t_{k\in\partial i\setminus j}}} \widetilde{\psi}_i(t_i, \mathbf{t_{k\in\partial i}}, \mathcal{O}_i, x_i^0) \prod_{k\in\partial_i\setminus j} m^{(t)}_{k\to i}(t_k,t_i)$;
    `Normalize` $m_{i\to j}$;
  end
end

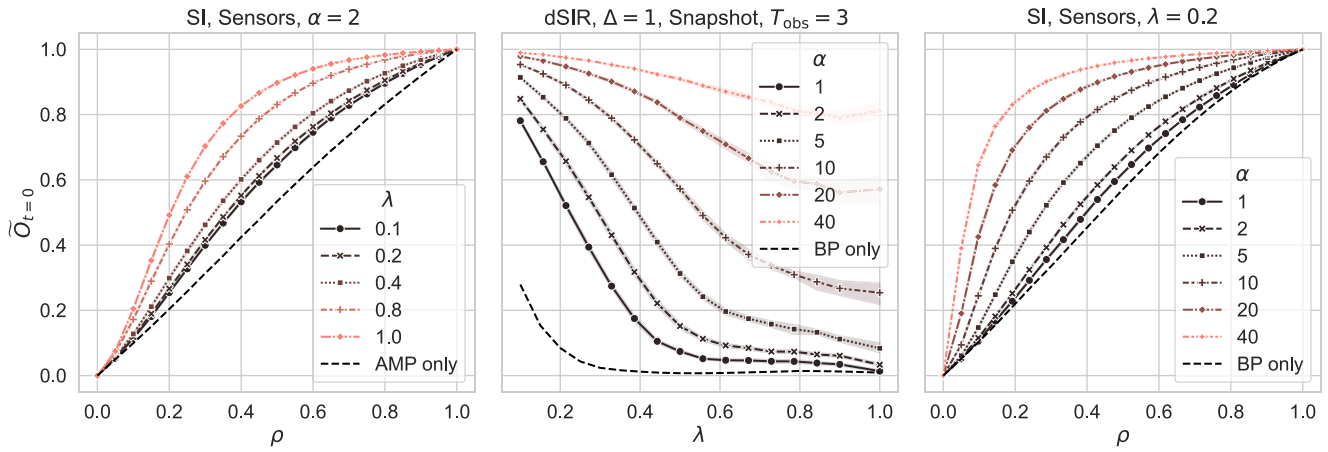

FIG. 2. Inference performance vs the transmission rate λ and the strength of correlations in the prior α. PSS model, Gaussian weights of the perceptron prior, $\kappa = -1$ ($\delta \approx 0.159$ sources). RRGs ($d = 3, N \times M = 1.6 \times 10^9$). Average of 20 runs; shading is 99% confidence interval. In the left and right panels, we take the case of sensor observations, showing the gain in performance when diminishing transmission rate λ and when increasing correlation in the prior through α, respectively. We also compare them to the baselines we defined in the text, AMP only and BP only, respectively. In the central panel, instead, the case of snapshot observations is considered, showing again a considerable increase in performance when using BP-AMP compared to only using BP.

We measure performance using a rescaled overlap for source retrieval:

$$\widetilde{\mathrm{O}}_{t=0} = \frac{\mathrm{O}_{t=0} - \mathrm{O}(\hat{\mathbf{x}}^{\mathbf{0,RND}}, \mathbf{x}^{*,\mathbf{0}})}{1 - \mathrm{O}(\hat{\mathbf{x}}^{\mathbf{0,RND}}, \mathbf{x}^{*,\mathbf{0}})}, \tag{29}$$

where $\hat{\mathbf{x}}^{\mathbf{0,RND}}$ is a random estimator, equivalent to the MMO estimator in Eq. (15) but without access to observations ($\mathcal{O} = \emptyset$). This rescaled overlap is non-negative and equals one at perfect recovery.

Figure 2 shows the impact of the infectivity of the epidemic process λ and the signal-to-noise ratio $\alpha = N/M$ on the algorithmic performance. Moreover, we compare the performance of the BP-AMP algorithm to two other algorithms:

(1) BP only: The node covariates F are not disclosed, and we rely only on the Belief Propagation for the inference (this corresponds to the $\alpha \to 0$ limit studied in [33]).

(2) AMP only: In the case of sensor observations, we can consider the case in which BP is not run on the graph, but we are only allowed to use the ρN observations as input variables for the AMP algorithm on the classical perceptron problem (this corresponds to the analysis done in [12]).

The left panel of Fig. 2 shows the rescaled overlap for the SI model with sensor observations versus the sensor fraction ρ. From this plot we can evince that lower stochasticity (higher λ) yields greater performance achieved from BP and equivalently a higher gain compared to the AMP-only case. Even at low λ, where spreading inference is harder, a significant performance gain is observed.

The central panel of Fig. 2 uses snapshot observations at fixed $T_{\text{obs}} = 3$ for dSIR ($\Delta = 1$). Here, inference quality depends on the outbreak size, controlled by λ. For small λ, the localized outbreak allows the snapshot to pinpoint sources, yielding near-perfect recovery. For large λ, most nodes are already infected and recovered, so the snapshot provides little topological information, limiting performance, especially for low α. Again, we notice an increased performance when using the information from the node covariates ($\alpha > 0$) compared to the BP-only case.

Finally, in the right panel we fix $\lambda = 0.2$, and we consider sensors in the SI model. By varying the sensors fraction ρ, we look at different values of α, so that we can compare again to the BP-only performance. We see also in this case that this baseline is recovered in the $\alpha \to 0$ limit, while for larger α the inference capability of the algorithm is greatly increased.

For the SI model without stochasticity ($\lambda = 1$), Fig. 3 shows performance versus sensor fraction ρ for varying α and κ. As expected, the overlap parameter increases with α, which acts as a signal-to-noise ratio, and the area under the curve increases monotonically as well. As $\alpha \to 0$, performance approaches that of the BP-only estimator (no neural-network prior). The smoothness of all curves confirms the absence of phase transitions in this setting.

VI. BINARY WEIGHTS PERCEPTRON PRIOR

We now analyze the case of binary weights of the perceptron prior using a Rademacher prior, i.e., $P^u(u_a) = \frac{1}{2}\delta_{u_a,+1} + \frac{1}{2}\delta_{u_a,-1}$. This choice reveals a different phenomenology compared to the Gaussian case, and we investigate its dependence on model parameters.

For concreteness, we fix the spreading model to be SI on a Random Regular Graph (RRG) with degree $d = 3$. As shown in Appendix C, these choices do not qualitatively affect the main results of this section. We use sensor observations, with the sensor fraction ρ acting as the information control parameter, and assess source retrieval via the overlap parameter in Eq. (29). Other inference tasks are deferred to Appendix C.

Before discussing the results, we comment on the Nishimori conditions. While the Gaussian case showed rapid convergence to validity of the Nishimori conditions necessary for Bayes optimality (Fig. 8), the Rademacher case exhibits significant finite-size effects near the perfect recovery transition we show below. These manifest in a small fraction of realizations where the mean overlap jumps to one at the fixed point, while the overlap remains finite. We show in Fig. 12 that this discrepancy diminishes with system size, and we

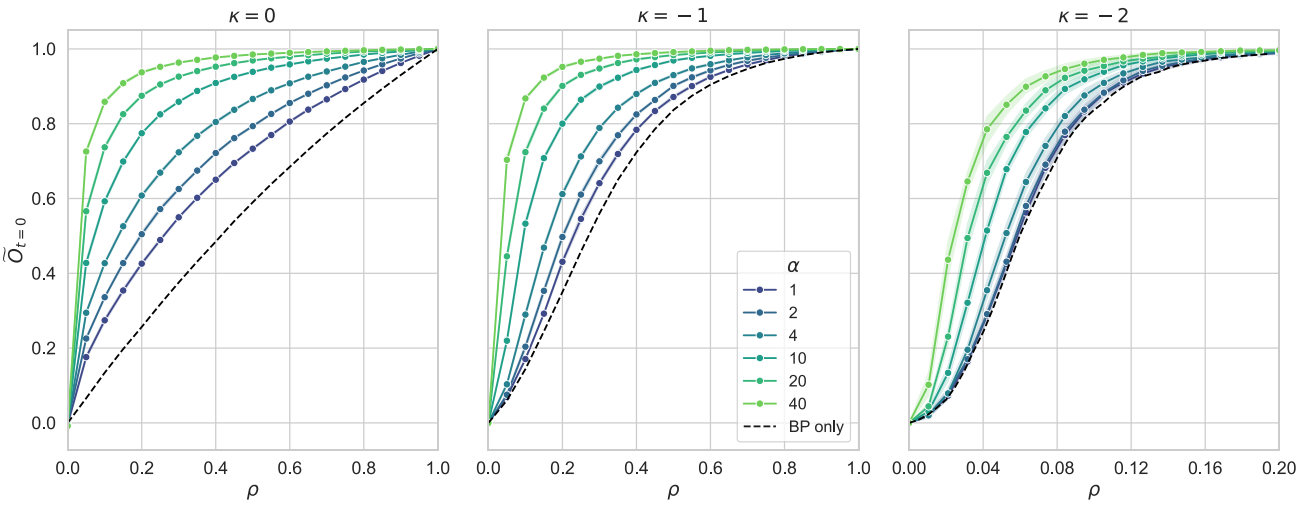

FIG. 3. Inference performance vs fraction of sensor observations. PSS model, Gaussian weights of the perceptron, varying correlation in the prior α, and density of sources κ (source fractions $\delta \approx 0.5, 0.159, 0.023$). SI model ($\lambda = 1$) with sensor observations. RRGs ($d = 3, N = 20\,000$). Rescaled overlap vs sensor fraction ρ. Average of 20 runs; shading is 99% confidence interval. The dashed lines are the performance of the BP algorithm, ignoring the part from the prior. We can see that BP-AMP improves substantially over this baseline, gaining more as the correlation between the initial states increases.

conjecture that it vanishes in the thermodynamic limit. In the plots below, we neglect these few atypical realizations, in order to better reflect the behavior of the model in typical instances.

Figure 4 shows the algorithm's performance for the non-stochastic SI model ($\lambda = 1$). Each panel plots the rescaled overlap $\widetilde{O}_{t=0}$ against sensor density ρ for a fixed κ and various ratios α. The dashed lines indicate the BP-only baseline. For small α, the curves follow the baseline. For intermediate and large α, we observe a sharp, discontinuous transition in the overlap, an effect more pronounced for more negative κ (sparser sources). This transition leads to perfect recovery ($\widetilde{O}_{t=0} = 1$) at a critical density $\rho_c(\alpha, \kappa)$. In the next subsection, we explore the algorithmic implications of this transition.

First-order phase transition and hardness

We now analyze the transition from Fig. 4 using the free entropy (Sec. IV). To probe the landscape for multiple stable states, we compare two initializations:

(1) Random: No observations ($\mathcal{O} = \emptyset$). BP messages are initialized uniformly. AMP variables are set as $\mathbf{a} \sim \mathcal{N}(0, 1/N)$ and $\mathbf{v} = \mathbf{1}$.

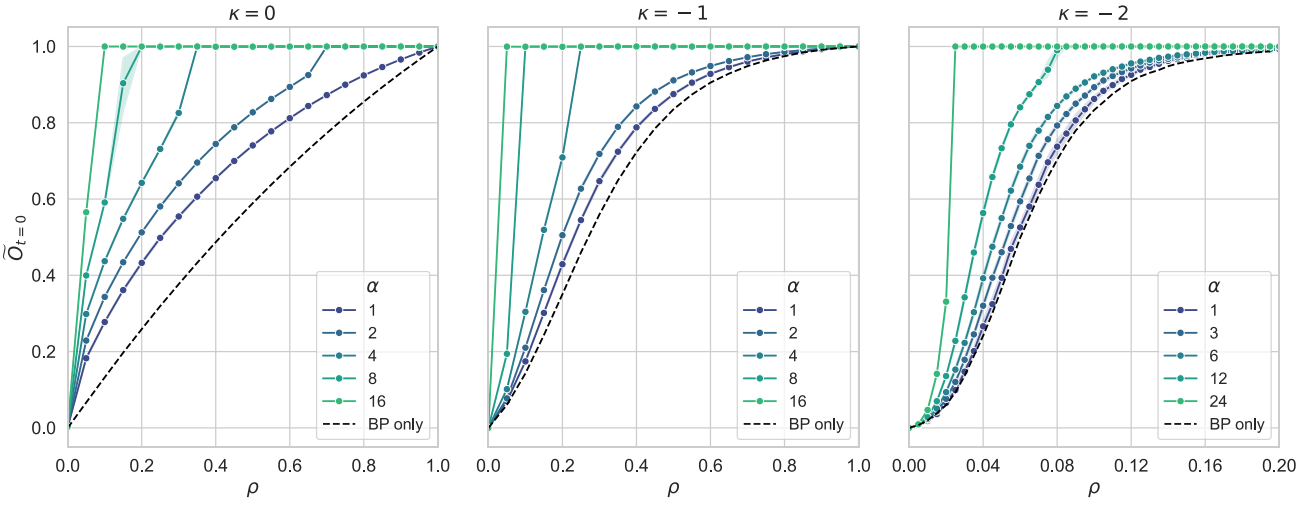

FIG. 4. Inference performance vs fraction of sensors for perceptron prior with Rademacher weights. PSS model, varying $\alpha = N/M$ and threshold $\kappa = 0, -1, -2$ (average source fractions $\delta \approx 0.5, 0.159, 0.023$). Sensor observations for SI model ($\lambda = 1$). Simulations on RRGs ($d = 3, N \times M = 1.6 \times 10^9$). Rescaled overlap in Eq. (29) vs sensor fraction ρ. Each point is an average of 20 runs starting from random initialization; shaded region is the 99% confidence interval. We observe a significant gain when using BP-AMP compared to BP alone. Compared to the Gaussian case in Fig. 3, where the curves remained continuous, here for $\alpha > \alpha_c$, studied in detail later, the algorithm achieves perfect recovery and the overlap jumps to one.

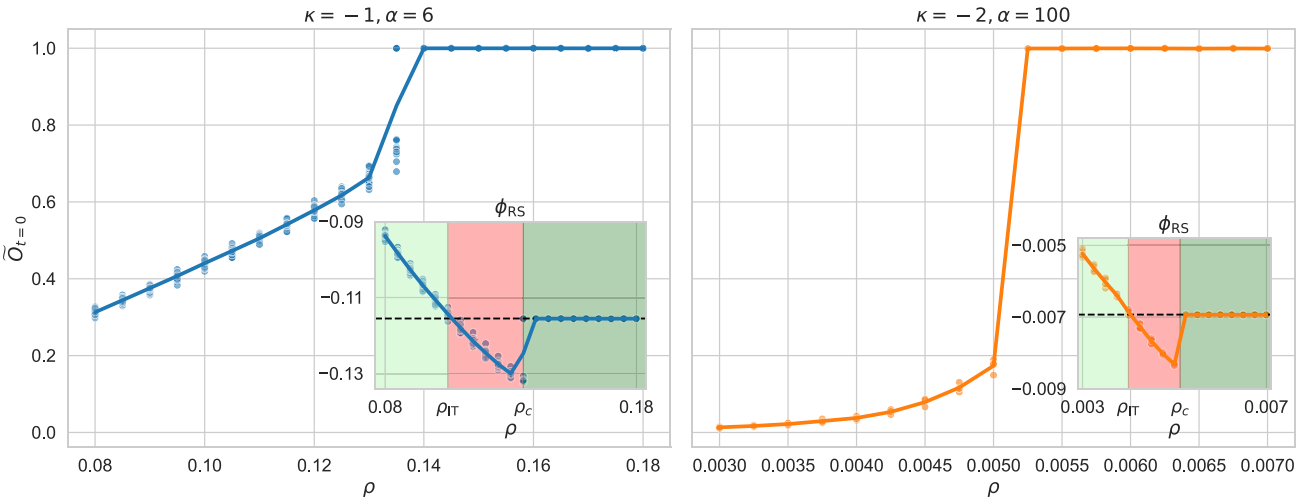

FIG. 5. Perfect recovery transition and Hard Phase. PSS model with Rademacher weights. Left: $\kappa = -1$ ($\delta \approx 0.159$), $\alpha = 6$. Right: $\kappa = -2$ ($\delta \approx 0.023$), $\alpha = 100$. SI model ($\lambda = 1$) with sensor observations. RRGs ($d = 3, N \times M = 2.5 \times 10^9$). In the main plots we show that the rescaled overlap has a jump when changing ρ, reaching perfect recovery for a certain ρ_c. At the same time, the free entropy (insets) shows the typical behavior of first-order transitions: the onset of the hard phase (shown in red) happens at ρ_{IT}, while at ρ_c there is perfect recovery and for all $\rho > \rho_c$ the free energy coincides with that of the informed fixed point (shown as a dashed black line).

(2) Informed: In addition to observations $\mathcal{O}$, the true initial state $\mathbf{x}^{*,\mathbf{0}}$ is provided. BP messages are uniform, but AMP variables are set to the ground truth $\mathbf{a} = \mathbf{u}$ and $\mathbf{v} = \mathbf{1}/\sqrt{N}$.

These initializations are designed to find two key fixed points: a "perfect recovery" state and a "partial recovery" state. The stability and free entropy of these fixed points characterize the first-order phase transition.

Using the sensor fraction ρ as the control parameter, we define ρ_c as the spinodal point where the partial recovery fixed point becomes unstable. For $\rho > \rho_c$, only the perfect recovery fixed point is stable, and the algorithm finds all sources, even from a random start. The perfect recovery fixed point is numerically stable for all $\rho \in [0, 1]$. Thus, for $\rho < \rho_c$ two scenarios arise based on which fixed point has a higher free entropy: (1) If the partial recovery fixed point is dominant, the algorithm's performance is Bayes optimal. (2) If the perfect recovery fixed point is dominant, the algorithm gets trapped in the metastable partial recovery state. This defines a computationally hard phase. The threshold between these cases is ρ_{IT} and if $\rho_{\mathrm{IT}} < \rho_c$ the hard phase exists for all $\rho \in [\rho_{\mathrm{IT}}, \rho_c]$.

Figure 5 illustrates this phenomenon for two (κ, α) pairs that exhibit a transition to perfect recovery ($\rho_c < 1$). As shown in Appendix B, the free entropy of the informed fixed point for $\lambda = 1$ is $\phi_{\mathrm{info}} = -\log 2/\alpha$. By comparing the free entropy from a random start, ϕ_{RS}, to ϕ_{info}, we compute the information-theoretic threshold ρ_{IT}. The insets show that for these parameters, a finite interval $\rho \in (\rho_{\mathrm{IT}}, \rho_c)$ forms a computationally hard phase: Bayes-optimal recovery is possible, but our algorithm fails to find it.

Figure 6 shows how the thresholds ρ_c and ρ_{IT} depend on α and λ. We compare these to the equivalent thresholds for the AMP-only perceptron problem (using ρN as input dimension), where then $\rho_c^{\mathrm{AMP}}(\kappa, \alpha) = \alpha_c^{\mathrm{AMP}}(\kappa)/\alpha$ and $\rho_{\mathrm{IT}}^{\mathrm{AMP}}(\kappa, \alpha) = \alpha_{\mathrm{IT}}^{\mathrm{AMP}}(\kappa)/\alpha$. The values for $\alpha_{c/\mathrm{IT}}^{\mathrm{AMP}}$ can be computed from State Evolution [12]. For high α, our BP-AMP algorithm requires a smaller sensor density for perfect recovery than pure AMP, and interestingly both follow a $\rho \sim 1/\alpha$ scaling. The curves converge at $\rho = 1$ and $\alpha = \alpha_{c/\mathrm{IT}}^{\mathrm{AMP}}$, since this corresponds to observing the whole system.

This behavior defines three distinct regions at fixed α, as ρ increases, which we illustrate more in detail in Appendix C 2:

(1) $\alpha < \alpha_{\mathrm{IT}}^{\mathrm{AMP}}$: The algorithm is Bayes Optimal but never achieves perfect recovery, as it is information-theoretically impossible.

(2) $\alpha_{\mathrm{IT}}^{\mathrm{AMP}} < \alpha < \alpha_c^{\mathrm{AMP}}$: A hard phase exists for $\rho > \rho_{\mathrm{IT}}^{\lambda}(\kappa)$. Perfect recovery is theoretically possible but algorithmically unreachable, as the spinodal $\rho_c > 1$.

(3) $\alpha > \alpha_c^{\mathrm{AMP}}$: All three regimes appear: Bayes optimal for $\rho < \rho_{\mathrm{IT}}^{\lambda}(\kappa)$, a hard phase for $\rho_{\mathrm{IT}}^{\lambda}(\kappa) < \rho < \rho_c^{\lambda}(\kappa)$, and finally perfect recovery for $\rho > \rho_c^{\lambda}(\kappa)$.

Looking at Fig. 6, we can also see how the spreading parameters affect these phases. A smaller source fraction (more negative κ) enhances the gain from BP, shifting the hard phase to lower sensor densities. Similarly, increasing stochasticity (smaller λ) decreases the gap with the AMP-only thresholds.

Finally, we comment on the relevance of these phase transitions to algorithmic hardness. While proving statements about all polynomial-time algorithms is beyond the scope of this work, significant results have established the asymptotic optimality of AMP among large classes of algorithms for various inference tasks [44,45]. Motivated by this line of research, we conjecture that our BP-AMP algorithm is also asymptotically optimal within these classes. Consequently, the hardness phenomena identified here likely represent fundamental computational hurdles.

VII. CONCLUSIONS

In this work, we introduced a model for local spreading on graphs where sources are generated by a neural-network prior rather than being sampled uniformly. Using statistical physics techniques, we derived a Bayesian inference algorithm that we conjecture to be asymptotically optimal among polynomial-time algorithms, when considering locally

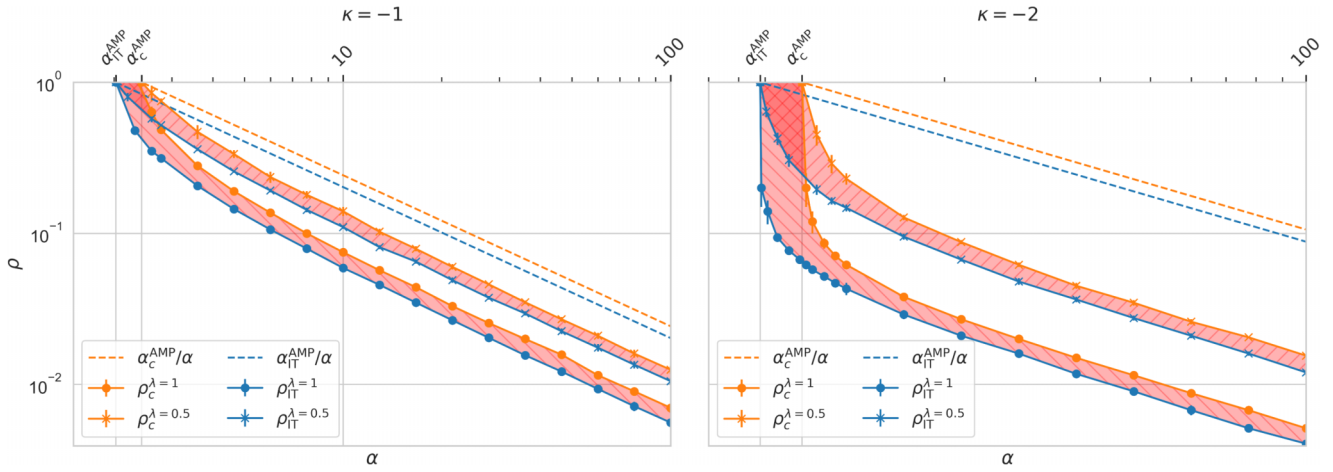

FIG. 6. First-order transition thresholds. PSS model with Rademacher weights. Left: $\kappa = -1$ ($\delta \approx 0.159$). Right: $\kappa = -2$ ($\delta \approx 0.023$). SI model ($\lambda \in \{0.5, 1\}$) with sensor observations. RRGs ($d = 3, N \times M \approx 2.5 \times 10^9$). Points show ρ_{IT} and ρ_c for BP-AMP, i.e., the information-theoretic threshold and the spinodal, respectively. Dashed lines are the same thresholds for the AMP-only problem, and we show how both move considerably to smaller values of ρ when introducing the information coming from BP, especially for low values of stochasticity (high values of λ).

treelike graphs and i.i.d. Gaussian features. We have analyzed the performance of the resulting BP-AMP algorithm against estimators that use only information from the spreading (BP only) or that use only the covariate variables (AMP only), finding a great advantage overall in combining the two techniques. For binary weights, our analysis revealed a statistical-to-computational gap, driven by first-order phase transitions to perfect recovery, as a function of the problem's signal-to-noise ratio. This phenomenology, absent in models with uniform priors, highlights how a neural-network model for the sources' prior can drastically alter the inference problem's hardness.

Several future directions stem from this work. We assumed all model parameters were known; a natural extension is to study the more realistic case where these parameters must be learned.

Future work could also explore more complex neural-network priors, such as multilayer architectures [9]. This would technically require combining results from [13] and [33], to study the influence of network depth. Alternatively, the neural prior could be applied directly to the transition times, which would necessitate a multiclass output AMP algorithm as derived in [46], or even to the model parameters, like the infection probabilities λ or the recovery delays Δ.

Finally, developing rigorous mathematical proofs for the more general hardness conjectures presented here remains an important open question.

ACKNOWLEDGMENTS

We thank Odilon Duranthon for useful discussions. This research was supported by the Swiss National Science Foundation (SNSF) under Grants No. 212049 (SMArtNet) and No. 200390 (OperaGOST).

DATA AVAILABILITY

The data that support the findings of this article are openly available [47].

APPENDIX A: DERIVATION OF ALGORITHM 1

We first review the cavity equations for the inference on the spreading process and for the neural-network prior separately, before showing how they are combined.

1. Belief propagation for the spreading process

This section reports the BP equations for spreading models with the probability function in Eq. (1). The BP equations for these models were first derived in [7]. We adopt the notation of [33], which is more directly adaptable to the present case.

The variables $\mathbf{T}_{(i,j)}$ in Fig. 1 compactly represent the pair of variables $\{\mathbf{t_i}, \mathbf{t_j}\}$ on a modified factor graph that preserves the local treelike structure. The detailed derivation can be found in [33]; here we give the final form of the message-passing equations:

$$m_{i\to j}(\mathbf{t_i}, \mathbf{t_j}) = \frac{1}{Z_{i\to j}} \sum_{x_i^0} \sum_{\{\mathbf{t_k}\}_{k\in\partial_i\setminus j}} \widetilde{\Psi}_i\big(\mathbf{t_i}, \{\mathbf{t_j}\}_{j\in\partial_i}, x_i^0, \mathcal{O}_i, \boldsymbol{\Theta}\big) \times \prod_{k\in\partial_i\setminus j} m_{k\to i}(\mathbf{t_k}, \mathbf{t_i}), \tag{A1}$$

from which one can compute the marginal for each variable as

$$b_i(\mathbf{t_i}) = \frac{1}{Z_i} \sum_{x_i^0} \sum_{\{\mathbf{t_k}\}_{k\in\partial_i}} \widetilde{\Psi}_i\big(\mathbf{t_i}, \{\mathbf{t_j}\}_{j\in\partial_i}, x_i^0, \mathcal{O}_i, \boldsymbol{\Theta}\big) \times \prod_{k\in\partial_i} m_{k\to i}(\mathbf{t_k}, \mathbf{t_i}) \tag{A2}$$

$$= \frac{Z_{i\to j}}{Z_i} \sum_{\mathbf{t_j}} m_{j\to i}(\mathbf{t_j}, \mathbf{t_i}) m_{i\to j}(\mathbf{t_i}, \mathbf{t_j}), \tag{A3}$$

where Z_i is the normalization. The free entropy for a single instance can then be written as

$$\frac{1}{N}\log Z = \frac{1}{N}\sum_{i=1}^{N}\log Z_i - \frac{1}{N}\sum_{(i,j)\in E}\log Z_{(i,j)}, \tag{A4}$$

where $Z_{(i,j)} = \sum_{\mathbf{t_i},\mathbf{t_j}} m_{j\to i}(\mathbf{t_j},\mathbf{t_i}) m_{i\to j}(\mathbf{t_i},\mathbf{t_j})$.

2. Approximate message passing for the Perceptron prior

AMP is a message-passing algorithm tailored for inference on dense graphical models, such as the Generalized Linear Models (GLMs) [11] that include the one-layer perceptron prior used here. We sketch its derivation, adapting the notation to our setting.

The GLM prior generates the initial states $\mathbf{x^0}$ from the vector of weights $\mathbf{u}\in\mathbb{R}^M$ (with prior P^u) and a random matrix $F\in\mathbb{R}^{N\times M}$ [with $F_{ia}\sim\mathcal{N}(0,1/M)$] via a perceptron channel:

$$P_{\text{out}}\left(x_i^0|z_i\right) = \delta\left(x_i^0 - \text{sgn}(z_i-\kappa)\right) \quad \text{where } z_i = \sum_{a=1}^{M} F_{ia}\,u_a. \tag{A5}$$

The factor graph for this prior is dense. In the thermodynamic limit [$M, N\to\infty$ with $\alpha = N/M = O(1)$], the cavity fields acting on any single variable are sums of many weakly correlated terms. By the central-limit theorem, these fields concentrate and their distributions become Gaussian, allowing them to be characterized by just their first two moments.

The key messages in the dense subgraph of Fig. 1 are $\chi_{i\to a}(u_a)$ (from node i to the weight a) and $\bar{\chi}_{a\to i}(u_a)$ (from weight a to node i). Due to the concentration argument, these messages can be approximated as Gaussian distributions. Specifically, the message $\bar{\chi}_{a\to i}(u_a)$ takes the form

$$\bar{\chi}_{a\to i}(u_a) \propto P^u(u_a)\exp\left[-\frac{A}{2}u_a^2 + B_{a\to i}u_a\right], \tag{A6}$$

where A and $B_{a\to i}$ are effective parameters representing the variance and mean of the cavity field on u_a. This Gaussian approximation is the foundation of the AMP algorithm.

The algorithm proceeds by iteratively updating estimates for the means (a_a) and variances (v_a) of the weights, and the effective fields (ω_i) acting on the output nodes. This is achieved through scalar denoising functions for the input (f_a, f_v) and output (g_o) channels. The standard AMP updates for a GLM take the form

$$V^{(t+1)} = \frac{1}{M}\sum_a v_a^{(t)}, \tag{A7}$$

$$\omega_i^{(t+1)} = \sum_a F_{ia}a_a^{(t)} - V^{(t+1)}g_{o,i}^{(t)}, \tag{A8}$$

$$g_{o,i}^{(t+1)} = g_o\left(\omega_i^{(t+1)}, x_i^0, V^{(t+1)}\right), \tag{A9}$$

$$A^{(t+1)} = \frac{1}{M}\sum_i \left(g_{o,i}^{(t+1)}\right)^2, \tag{A10}$$

$$B_a^{(t+1)} = \sum_i F_{ia}g_{o,i}^{(t+1)} + a_a^{(t)}A^{(t+1)}, \tag{A11}$$

$$a_a^{(t+1)} = f_a\left(A^{(t+1)}, B_a^{(t+1)}\right), \quad v_a^{(t+1)} = f_v\left(A^{(t+1)}, B_a^{(t+1)}\right). \tag{A12}$$

Here, the terms $g_{o,i}^{(t)}$ in the update for ω_i and $a_a^{(t)}A^{(t+1)}$ in the update for B_a are the crucial Onsager reaction terms. They correct for the "self-interaction" that would otherwise arise from using a variable's estimate to compute the field acting back on it, ensuring the algorithm's dynamics can be tracked accurately by State Evolution [12]. The functions g_o, f_a, and f_v are the scalar denoisers defined in the main text.

3. Putting the two together

In the NSS model, the posterior factorizes into two subproblems: a dense GLM part coupling $(\mathbf{u}, \mathbf{x}^0)$ and a sparse spreading part coupling $(\mathbf{x}^0, \{\mathbf{t}_i\})$. This structure motivates a hybrid message-passing architecture: AMP handles the dense GLM part, providing prior messages $\eta_i(x_i^0)$ to BP, which in turn computes refined beliefs $\nu_i(x_i^0)$ that act as effective likelihoods for AMP.

Concretely, at each iteration t we perform two steps:

(1) BP step: Update all epidemic messages $m_{i\to j}^{(t)}(\mathbf{t}_i,\mathbf{t}_j)$ using Eq. (A1), where the source term is now weighted by the message $\eta_i^{(t)}(x_i^0)$ from AMP. From the updated messages, compute the new marginals on the initial states:

$$\nu_i^{(t+1)}\left(x_i^0\right) = \sum_{t_i,\mathbf{t}_{\mathbf{k}\in\partial\mathbf{i}}} \widetilde{\psi}_i\left(t_i,\mathbf{t}_{\mathbf{k}\in\partial\mathbf{i}},\mathcal{O}_i,x_i^0\right)\prod_{k\in\partial i} m_{k\to i}^{(t)}(t_k,t_i). \tag{A13}$$

(2) AMP step: Use the BP marginals $\nu_i^{(t+1)}(x_i^0)$ as effective, data-dependent likelihoods in the GLM-AMP update. The resulting field, to be passed back to BP, is computed as

$$\eta_i^{(t+1)}\left(x_i^0\right) \propto \int \mathrm{d}z\, P_{\text{out}}\left(x_i^0|z\right)\exp\left[-\frac{\left(z-\omega_i^{(t+1)}\right)^2}{2V^{(t+1)}}\right]. \tag{A14}$$

To merge the two algorithms, we only need to modify the output denoising function g_o to incorporate the prior knowledge $\nu_i(x_i^0)$ from the BP step:

$$g_o(\omega,\nu,V) = \frac{\int \mathrm{d}z\sum_{x^0}\nu(x^0)P_{\text{out}}(x^0|z)(z-\omega)e^{-[(z-\omega)^2/2V]}}{V\int \mathrm{d}z\sum_{x^0}\nu(x^0)P_{\text{out}}(x^0|z)e^{-[(z-\omega)^2/2V]}}. \tag{A15}$$

With this modification, the AMP updates in Eq. (A7) and the BP updates in Eq. (A1) (now weighted by η_i) are iterated, forming the complete Algorithm 1. This construction is inspired by [14,33]. The limit $F\to 0$ recovers the BP-only inference from [33], while the limit $\lambda_{ij}\to 0$ collapses to the standard perceptron AMP from [11].

In practice, for all experiments we have used damping to ensure convergence across all regimes. Furthermore, we have used the following schedule, which proved empirically to be the fastest: at each update of the BP messages, through Eq. (A1), we run the AMP algorithm until convergence of the AMP variables. Then we fix these variables and run the next BP update, and so on until convergence of the BP messages. It could be interesting to explore different schedules to see how merging AMP and BP in different ways changes the convergence time of the total algorithm.

APPENDIX B: FREE ENTROPY

Given a probability distribution, and the associated factor graph, the Bethe approximation of the log-partition function is given by

$$\log Z_{\text{Bethe}} = \sum_a \phi_a - \sum_{(a,b)} \phi_{ab}, \tag{B1}$$

where the sum over a runs over all factor nodes and variable nodes, and (a, b) runs over all edges in the factor graph. We can decompose the free entropy into three pieces:

$$\phi_{\text{RS}} \equiv \frac{1}{N}\log Z_{\text{Bethe}} = \phi_{\text{spread}} + \phi_{\text{GLM}} + \phi_{\text{inter}}, \tag{B2}$$

with

$$\phi_{\text{spread}} = \frac{1}{N}\sum_{i=1}^{N}\phi_i^{\text{spread}} + \frac{1}{N}\sum_{(i,j)\in E}\phi_{(i,j)}^{\text{spread}} - \frac{1}{N}\sum_{(i,j)\in E}\phi_{i,(i,j)}^{\text{spread}} - \frac{1}{N}\sum_{(i,j)\in E}\phi_{j,(i,j)}^{\text{spread}}, \tag{B3}$$

$$\phi_{\text{GLM}} = \frac{1}{N}\sum_{i=1}^{N}\phi_i^{\text{out}} + \frac{1}{N}\sum_{a=1}^{M}\phi_a - \frac{1}{N}\sum_{i=1}^{N}\sum_{a=1}^{M}\phi_{ia}, \tag{B4}$$

$$\phi_{\text{inter}} = \frac{1}{N}\sum_{i=1}^{N}\phi_i^0 - \frac{1}{N}\sum_{i=1}^{N}\phi_i^{0,\text{out}} - \frac{1}{N}\sum_{i=1}^{N}\phi_i^{0,\text{spread}}. \tag{B5}$$

As done in [33], one can show that $\phi_{i,(i,j)}^{\text{spread}} = \phi_{j,(i,j)}^{\text{spread}} = \phi_{(i,j)}^{\text{spread}}$, so that for the spreading part we have

$$\phi_{\text{spread}} = \frac{1}{N}\sum_{i=1}^{N}\phi_i^{\text{spread}} - \frac{1}{N}\sum_{(i,j)\in E}\phi_{(i,j)}^{\text{spread}}, \tag{B6}$$

where

$$\phi_i^{\text{spread}} = \log \sum_{x_i^0, t_i, \{t_j\}_{j\in\partial i}} \eta_i(x_i^0)\,\widetilde{\psi}_i\big(x_i^0, t_i, \{t_j\}_{j\in\partial i}, \mathcal{O}_i, \boldsymbol{\lambda}, F_i, \mathbf{u}\big) \times \prod_{j\in\partial i} m_{j\to i}(t_j, t_i) \tag{B7}$$

and

$$\phi_{(i,j)}^{\text{spread}} = \log \sum_{t_i,t_j} m_{i\to j}(t_i, t_j) m_{j\to i}(t_j, t_i). \tag{B8}$$

By the same argument, one can show that $\phi_i^0 = \phi_i^{0,\text{out}} = \phi_i^{0,\text{spread}}$ and thus

$$\phi_{\text{inter}} = -\frac{1}{N}\sum_{i=1}^{N}\phi_i^0, \tag{B9}$$

in which

$$\phi_i^0 = \log \sum_{x_i^0} \eta_i(x_i^0)\nu_i(x_i^0) \tag{B10}$$

$$= \log\left(\frac{1}{Z_i^\nu}\sum_{x_i^0}\eta_i(x_i^0)\sum_{t_i,\mathbf{t}_{\mathbf{k}\in\partial\mathbf{i}}}\widetilde{\Psi}_i\big(t_i, \mathbf{t}_{\mathbf{k}\in\partial\mathbf{i}}, x_i^0, \mathcal{O}_i\big) \times \prod_{k\in\partial i} m_{k\to i}(t_k, t_i)\right) \tag{B11}$$

$$= \phi_i^{\text{spread}} - \log Z_i^\nu, \tag{B12}$$

where we made the message ν_i explicit. It is easy to see that in the total free entropy in Eq. (B2), the first term in Eq. (B12) cancels the first term in Eq. (B6).

Finally, for the GLM part we have

$$\phi_a = \log\left(\int \mathrm{d}u_a \psi_a(u_a)\prod_j \chi_{j\to a}(u_a)\right) \tag{B13}$$

$$= \log\left(\int \mathrm{d}u_a\psi_a(u_a)\prod_j\left[\frac{1}{Z_{j\to a}}\sum_{x_j^0}\nu_j(x_j^0) \times \int\left(\prod_{b\neq a}\mathrm{d}u_b\right)\Psi_j^{\text{out}}(x_j^0, \mathbf{u}, \mathbf{F_j})\prod_{b\neq a}\bar{\chi}_{b\to j}(u_b)\right]\right) \tag{B14}$$

$$= -\sum_j \log Z_{j\to a} + \log Z_a \tag{B15}$$

and

$$\phi_i^{\text{out}} = \log\left(\sum_{x_i^0}\nu_i(x_i^0)\int\left(\prod_b \mathrm{d}u_b\right)\Psi_i^{\text{out}}(x_i^0, \mathbf{u}, \mathbf{F_i}) \times \prod_b \bar{\chi}_{b\to i}(u_b)\right), \tag{B16}$$

while

$$\phi_{ia} = \log\left(\int \mathrm{d}u_a \bar{\chi}_{a\to i}(u_a)\chi_{i\to a}(u_a)\right) \tag{B17}$$

$$= \log\left(\frac{1}{Z_{i\to a}}\sum_{x_i^0}\nu_i(x_i^0)\int\left(\prod_b \mathrm{d}u_b\right)\Psi_i^{\text{out}}(x_i^0, \mathbf{u}, \mathbf{F_i}) \times \prod_b \bar{\chi}_{b\to i}(u_b)\right) \tag{B18}$$

$$= \phi_i^{\text{out}} - \log Z_{i\to a}. \tag{B19}$$

Putting this all together, we get

$$\phi_{\text{RS}} = \frac{1}{N}\sum_{i=1}^{N}\log Z_i^\nu - \frac{1}{N}\sum_{(i,j)\in E}\phi_{(i,j)}^{\text{spread}} + \frac{1}{N}\sum_{a=1}^{M}\log Z_a + \frac{1-M}{N}\sum_{i=1}^{N}\phi_i^{\text{out}}. \tag{B20}$$

One can show (see, e.g., [14], Appendix B]) that

$$\sum_{a=1}^{M}\log Z_a = M\sum_{i=1}^{N}\phi_i^{\text{out}} + \sum_{a=1}^{M}\log\int \mathrm{d}u_a P^u(u_a) \times \exp\left[-\frac{A_a}{2}u_a^2 + B_a u_a\right] + \sum_{a=1}^{M}\left[\frac{A_a}{2}(a_a^2 + v_a) - B_a a_a\right] + \sum_{i=1}^{N}\frac{\left(\omega_i - \sum_{a=1}^{M}F_{ia}a_a\right)^2}{2V_i}. \tag{B21}$$

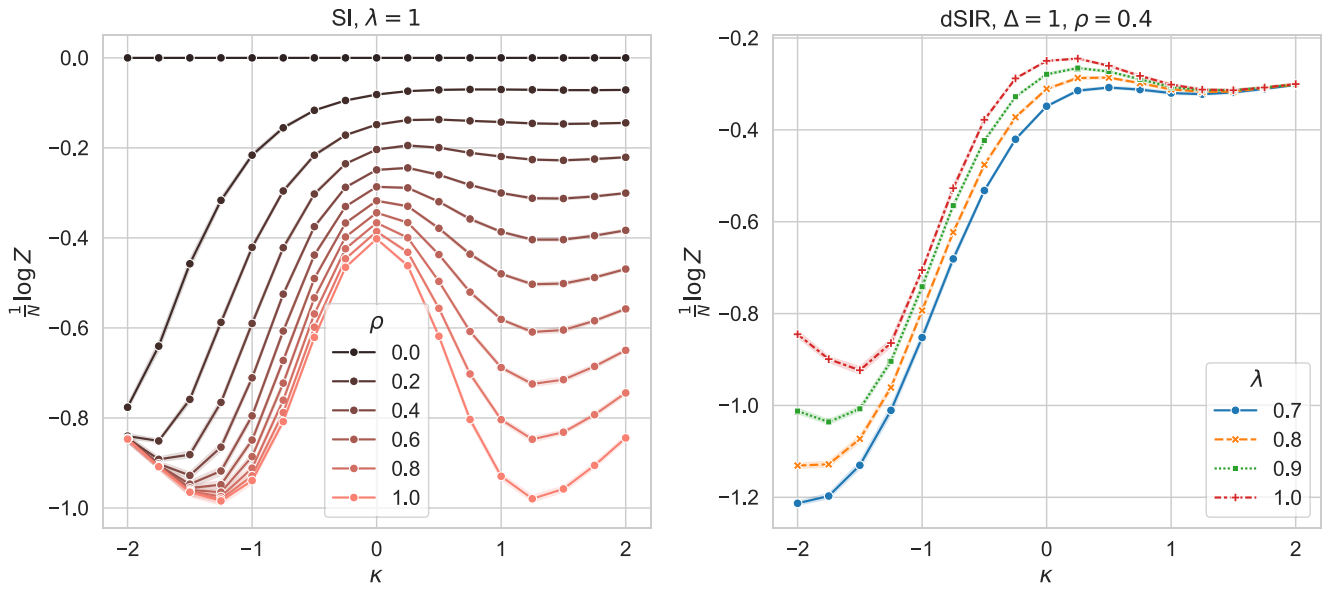

FIG. 7. Free entropy with Gaussian weights. PSS model, Gaussian weights, $\alpha = 4$. The x axis varies the threshold κ, changing the source fraction δ. We plot the total free entropy for sensor observations. Left: SI model ($\lambda = 1$) with varying sensor fraction ρ. Right: dSIR model ($\Delta = 1$) with fixed $\rho = 0.4$ and varying transmission probability λ. Simulations on RRGs ($d = 3, N = 20\,000$). Each point is an average of 20 runs; shading is the 99% confidence interval. Results from random and informed initializations are indistinguishable.

To conclude, the final formula for the RS free entropy reads

$$\begin{aligned}\phi_{\rm RS} = &\frac{1}{N}\sum_{i=1}^{N}\log Z_i^v - \frac{1}{N}\sum_{(i,j)\in E}\phi_{(i,j)}^{\rm spread} \\ &+ \sum_{a=1}^{M}\log\int {\rm d}u_a P^u(u_a)\exp\left[-\frac{A_a}{2}u_a^2 + B_a u_a\right] \\ &+ \frac{1}{N}\sum_{i=1}^{N}\phi_i^{\rm out} + \sum_{a=1}^{M}\left[\frac{A_a}{2}\left(a_a^2 + v_a\right) - B_a\, a_a\right] \\ &+ \sum_{i=1}^{N}\frac{\left(\omega_i - \sum_{a=1}^{M}F_{ia}a_a\right)^2}{2V_i},\end{aligned} \tag{B22}$$

and it can be computed in linear time by using the BP messages and AMP variables in Algorithm 1.

Figure 7 shows the free entropy for the Gaussian weights case, where no first-order phase transitions are observed. The left panel plots the free entropy as a function of the perceptron threshold κ, which controls the source density. While the profile is symmetric around $\kappa = 0$ for $\rho = 0$ (no observations) and $\rho = 1$ (full observation), it becomes asymmetric for intermediate sensor fractions. The right panel shows that spreading stochasticity (varying λ) has a relatively minor impact on the free entropy, even for the dSIR model with $\Delta = 1$, where its effect should be most pronounced.

Finally, we discuss the calculation of the free entropy at the informative fixed point, $\phi_{\rm info}$. For the nonstochastic SI model ($\lambda = 1$), recovering the sources is equivalent to recovering the full infection trajectories. This corresponds to a state where all BP messages are delta functions centered on the ground-truth values. In this simplified scenario, the only nontrivial contribution to the free entropy comes from the prior over the weights, yielding

$$\phi_{\rm info}(\lambda = 1) = \frac{1}{\alpha}\mathbb{E}_u \log P^u(u). \tag{B23}$$

For the stochastic case ($\lambda \neq 1$), the situation is more complex, as perfect source recovery no longer implies perfect trajectory recovery. To compute $\phi_{\rm info}$ in this setting, we run the algorithm with an additional observation of the true initial state for all nodes. The algorithm then infers the subsequent spreading given the known sources, and the free entropy at the resulting fixed point gives the value of $\phi_{\rm info}$.

APPENDIX C: ADDITIONAL PLOTS

1. Nishimori conditions for Gaussian weights

In this section, we check that the Nishimori conditions are satisfied in the case of Gaussian weights. In Fig. 8, we fix $\kappa = -1$, so that on average the fraction of sources is $\delta = \frac{1}{2}[1 - \mathrm{erf}(1/\sqrt{2})] \approx 0.159$. We then explore the parameter space of our problem, and we find that in all cases, on average, the Nishimori conditions are satisfied. We plot the curves for different values of the system size N and, as expected from self-averaging properties, we see that the difference between the overlap and the mean overlap diminishes when we increase N.

2. Phenomenology of the first-order transition for Rademacher variables

In this section, we present an example of what happens when changing the ratio α in the phase diagram for Rademacher variables, presented at the end of Sec. VI. Specifically, in Fig. 9 we take the case $\kappa = -2$ and $\lambda = 0.5$, and we show the behavior of the overlap and Free energy as a function of ρ for three cases: (1) $\alpha = 5 < \alpha_{\rm IT}^{\rm AMP}$, in which

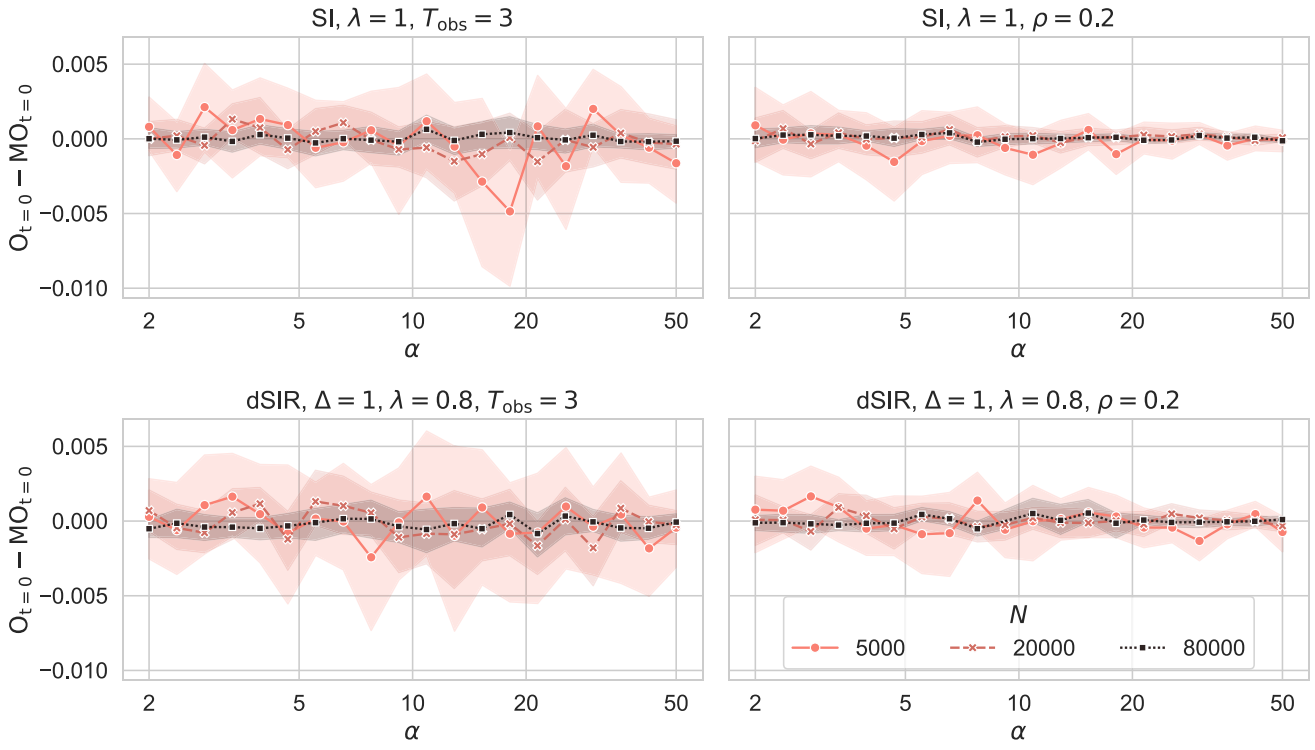

FIG. 8. Check of the Nishimori conditions. We consider the PSS model with $\kappa = -1$ and Gaussian weights. We plot the difference between overlap and mean overlap at $t = 0$ as a function of $\alpha \in [2,48]$, in logarithmic scale. In the left panels, we consider snapshot observations at time $T_{\text{obs}} = 3$, while on the right we look at the case in which there is a fraction $\rho = 0.2$ of sensors. At the same time, in the upper panel we look at the SI model with transmission parameter $\lambda = 1$, while in the lower panels we consider the dSIR model with $\Delta = 1$ and $\lambda = 0.8$. All simulations are done on RRGs of degree $d = 3$, and we vary the system size $N \in [5000, 80\,000]$ to check for finite-size effects. Each point in the plot has been computed by averaging 20 different simulations, and the shaded region represents the 99% confidence interval.

there is no hard phase and no transition, and thus the algorithm is always Bayes optimal. (2) $\alpha_{\text{IT}}^{\text{AMP}} < \alpha = 10 < \alpha_c^{\text{AMP}}$, where again there is no transition to perfect recovery but now for $\rho > \rho_{\text{IT}}$ we see a hard phase by looking at the free energy. (3) $\alpha > \alpha_c^{\text{AMP}}$, in which we finally have a transition to perfect recovery for $\rho = \rho_c$.

3. Mean-squared error

In the main text, we focused on the problem of retrieving the sources of the spreading, for convenience. However, using a Bayesian approach allows us to characterize the performance of our algorithm in very generic tasks, just by using the marginals estimated through the belief propagation and approximate message-passing algorithms. For example, it is interesting to look at how well the algorithm is able to recover the entire trajectory of each individual, which for $\lambda < 1$, where the spreading is stochastic, is a problem strictly harder than source recovering. If we restrict to models in which there is a single transition time, like the SI and dSIR models we described in the main text, we can use as performance parameter the Squared Error between the ground-truth transmission times $\mathbf{t^*} \equiv \{t_i^*\}_{i=1}^N$ and the ones estimated through the algorithm $\hat{\mathbf{t}} \equiv \{\hat{t}_i\}_{i=1}^N$, defined as

$$\text{SE}(\mathbf{t^*}, \hat{\mathbf{t}}) \equiv \frac{1}{N}\sum_{i=1}^{N}(\hat{t}_i - t_i^*)^2. \tag{C1}$$

As for the overlap estimator, we assume that the algorithm has additional information on the process through partial observations and feature covariates, so that the Bayes-optimal estimator uses the posterior probability distribution $P(\mathbf{t}|\boldsymbol{\mathcal{O}}, F)$ and evaluates the Mean-Squared Error, defined as

$$\begin{aligned}\text{MSE}(\hat{\mathbf{t}}) &\equiv \mathbb{E}_{\mathbf{t}|\boldsymbol{\mathcal{O}},F}[\text{SE}(\mathbf{t}, \hat{\mathbf{t}})] \\ &= \frac{1}{N}\sum_{\mathbf{t}} P(\mathbf{t}|\boldsymbol{\mathcal{O}}, F)\sum_{i=1}^{N}(\hat{t}_i - t_i)^2.\end{aligned} \tag{C2}$$

In this case, in order to maximize this quantity over $\hat{\mathbf{t}}$, one can show [11] that we have to consider the minimum mean-squared error estimator:

$$\begin{aligned}\hat{t}_i^{\text{MMSE}} &= \sum_{\mathbf{t}} t_i P(\mathbf{t}|\boldsymbol{\mathcal{O}}, F) = \sum_{t_i} t_i P_i(t_i|\boldsymbol{\mathcal{O}}, F) \\ &\forall i \in [1:N],\end{aligned} \tag{C3}$$

where we have defined the marginal probability $P_i(t_i|\boldsymbol{\mathcal{O}}, F) \equiv \sum_{\{t_j\}_{j\neq i}} P(\mathbf{t}|\boldsymbol{\mathcal{O}}, F)$. In this section, we will use as performance parameters the rescaled squared error

$$R_{\text{SE}} = \frac{\text{SE}(\mathbf{t^*}, \hat{\mathbf{t}}^{\text{RND}}) - \text{SE}(\mathbf{t^*}, \hat{\mathbf{t}}^{\text{MMSE}})}{\text{SE}(\mathbf{t^*}, \hat{\mathbf{t}}^{\text{RND}})} \tag{C4}$$

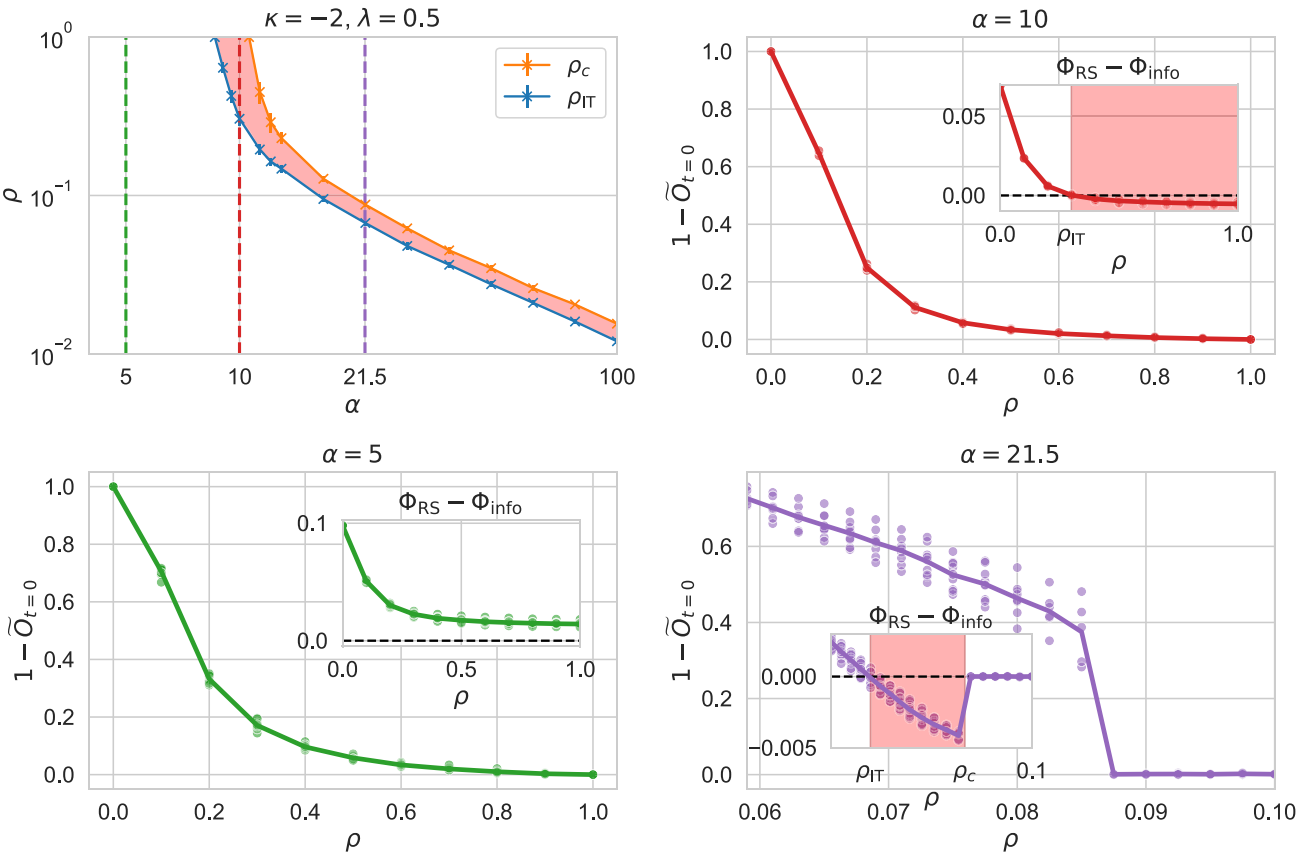

FIG. 9. Phase transition phenomenology. We fix $\kappa = -2$ (corresponding to $\delta \approx 0.023$) and $\lambda = 0.5$. Top left: (ρ, α) phase diagram. Other plots show free energy difference and $1 - \widetilde{O}_{t=0}$ vs ρ for fixed α in each of the three regimes. We see that, from low to high α, we first have no phase transition, then we encounter a hard phase but do not have perfect recovery, and finally we have both the hard phase and the perfect recovery phase.

and the associated rescaled mean-squared error

$$R_{\mathrm{MSE}} = \frac{\mathrm{MSE}(\hat{\mathbf{t}}^{\mathrm{RND}}) - \mathrm{MSE}(\hat{\mathbf{t}}^{\mathrm{MMSE}})}{\mathrm{MSE}(\hat{\mathbf{t}}^{\mathrm{RND}})}, \tag{C5}$$

where $\hat{t}_i^{\mathrm{RND}} = \sum_{t_i} P_i(t_i|\emptyset, F)t_i$ is the random estimator.

Notice that these parameters are equal to zero if the MMSE estimator and the RND estimator have the same performance, and they are equal to 1 if the MMSE estimator recovers all the trajectories perfectly. As done for the overlap, we can probe the optimality of the algorithm by checking the Nishimori conditions, i.e., the difference between R_{SE} and R_{MSE}, since from the Nishimori conditions we know that they coincide on average in the Bayes-optimal setting. In Figs. 10 and 11 we show the performance of the algorithm in terms of the squared error, in the same settings as Figs. 3 and 4 in the main text. We see again that for α going to zero we retrieve the performance of BP, while the more we increase the correlations between the weights (by increasing α) the more the performance develops a gap from the BP-only baseline.

4. Finite-size study for Rademacher variables

In this section, we discuss the behavior of the Nishimori conditions when considering Rademacher variables and values of ρ near the transition to perfect recovery.

In Fig. 12, we consider the setting analyzed in Sec. VI, looking at the SI model with $\lambda = 1$, and fixing as an example $\alpha = 10$ and $\kappa = 0$. We compute the values of the rescaled overlap and mean overlap as a function of ρ, looking at different sizes N. We see that the more we increase N, the more the transition gets sharp, signifying that the model presents strong finite-size corrections even at moderately large sizes. In the inset plot we look at the difference between overlap and mean overlap, to investigate the Nishimori conditions. We see that taking N larger shrinks the region where the conditions are not satisfied, leading us to conjecture that in the large N limit, when the transition is very sharp, the Nishimori conditions are satisfied for all values of ρ.

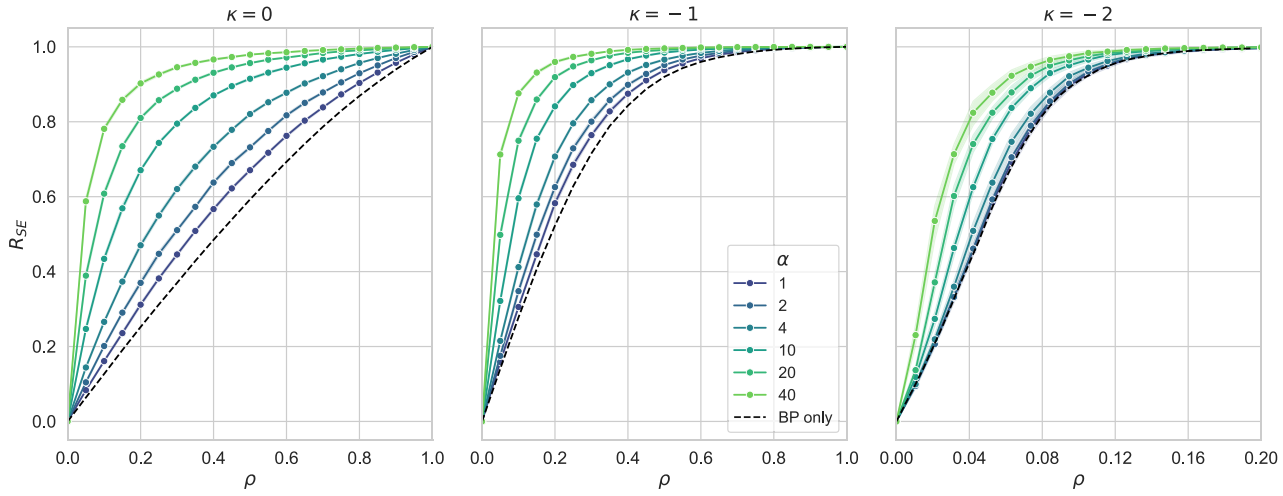

FIG. 10. Squared Error (Gaussian). We look at the PSS model with Gaussian weights, varying the ratio $\alpha = N/M$. We also choose the threshold value $\kappa = 0, -1, -2$, corresponding to an average fraction of sources $\delta \approx 0.5, 0.159, 0.023$, respectively. We consider observations through sensors, considering the SI model with $\lambda = 1$. All simulations are done on RRGs of degree $d = 3$ and size $N = 20\,000$. We plot the rescaled squared error defined in Eq. (C4) as a function of the fraction of sensors $\rho \in [0, 1]$. Each point in the plot has been computed by averaging 20 different simulations, and the shaded region represents the 99% confidence interval.

5. Other graph ensembles

In the main text, we have fixed the ensemble of graphs in which the spreading happens to be RRGs of degree $d = 3$, for convenience. Here we present results for some other variants of locally treelike random graphs, to show that the phenomenology we describe in the paper remains unchanged. Specifically, we change the degree of the graphs to $d = 5$, and we compare the ensemble of RRGs to ER graphs [48], where each couple of nodes is randomly connected with probability p. To make the two ensembles of graphs comparable, we set $p = \frac{d}{N-1}$, such that in both cases the average degree of the nodes is d. Moreover, if the graph is composed by multiple disconnected components, we consider only the biggest one and neglect the others (for $d = 5$ and N large enough, the largest component comprehends almost all the nodes with high probability).

In Fig. 13 we look at both the overlap at time $t = 0$ and the squared error, considering the PSS model with Gaussian

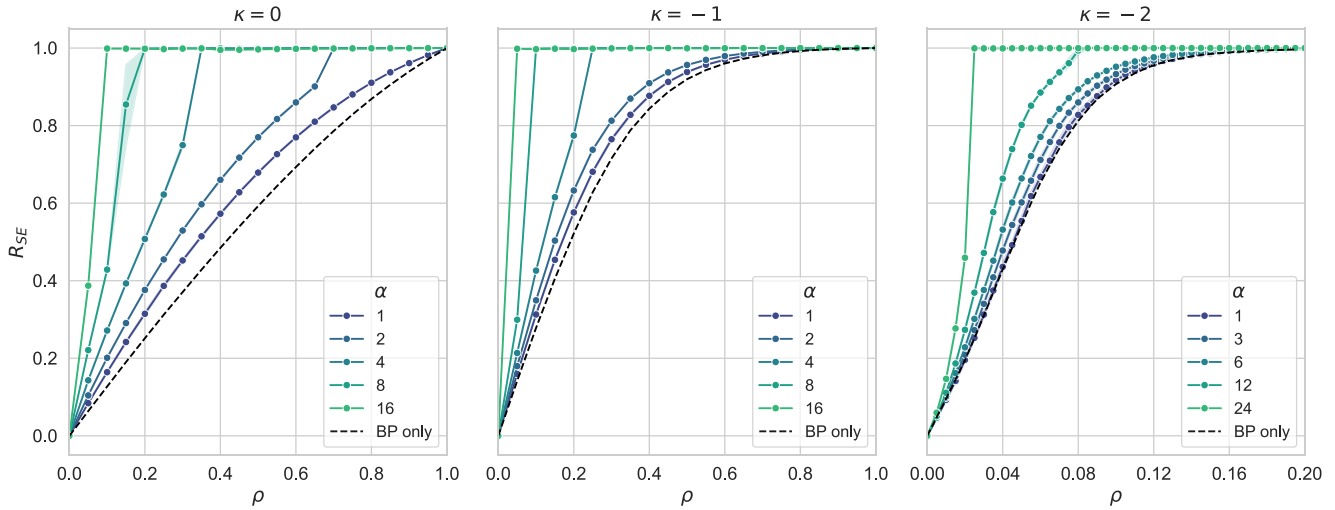

FIG. 11. Squared Error (Rademacher). We look at the PSS model with Rademacher weights, varying the ratio $\alpha = N/M$. We also choose $\kappa = 0, -1, -2$, corresponding to an average fraction of sources $\delta \approx 0.5, 0.159, 0.023$, respectively. We consider observations through sensors, considering the SI model with $\lambda = 1$. All simulations are done on RRGs of degree $d = 3$ and fixing the size N in such a way that $M * N = 1.6 \times 10^9$. We plot the rescaled squared error defined in Eq. (C4) as a function of the fraction of sensors $\rho \in [0, 1]$. Each point in the plot has been computed by averaging 20 different simulations, and the shaded region represents the 99% confidence interval.

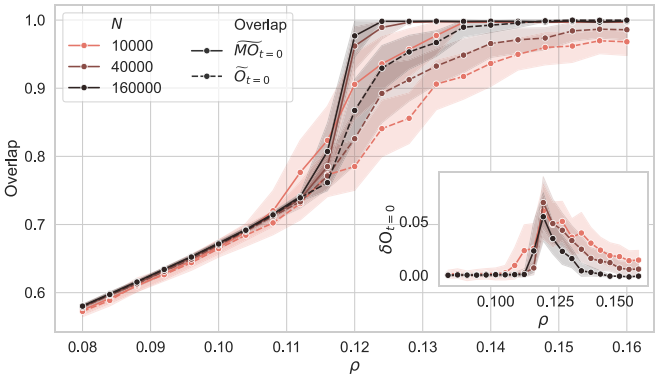

FIG. 12. Nishimori conditions and finite sizes. We consider the PSS model with Rademacher weights, fixing $\alpha = 10$ and the threshold constant $\kappa = 0$. The spreading model is fixed to be SI with a transmission parameter $\lambda = 1$, on RRGs of average degree $d = 3$. Varying ρ on the x axis, on the main panel we compare the rescaled overlap to the rescaled mean overlap, both at time $t = 0$, while in the inset panel we plot the difference between the two. Furthermore, we look at different values of N, to study how the behavior changes when increasing the size of the system. Each point in the plot has been computed by averaging 25 different simulations, and the shaded region represents the 99% confidence interval.

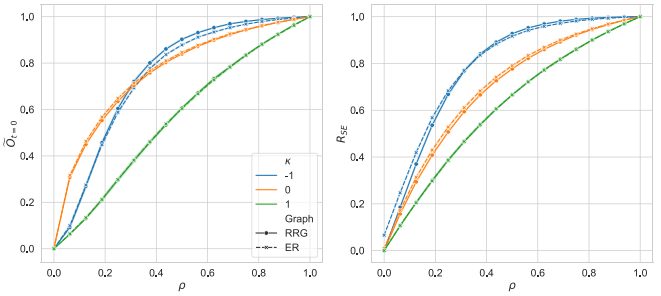

FIG. 13. Comparison between Ensembles. We consider the PSS model with Gaussian weights, fixing $\alpha = 4$ and changing the threshold constant $\kappa = -1, 0, 1$, corresponding to an average fraction of sources $\delta \approx 0.841, 0.5, 0.159$, respectively. The spreading model is fixed to be SI with a transmission parameter $\lambda = 1$, and we compare RRGs and ER graphs, fixing the number of nodes $N = 20\,000$ and the average degree $d = 5$ as explained in the text. In the left panel we plot the rescaled overlap at time $t = 0$ defined in Eq. (29), while in the right the rescaled squared error defined in Eq. (C4), both as a function of the fraction of sensors ρ. Each point in the plot has been computed by averaging 20 different simulations, and the shaded region represents the 99% confidence interval.

weights and varying threshold κ. We show that the choice of the ensemble only slightly impacts the inference capabilities of our algorithm, which justifies our choice of fixing the graph's ensemble in the main text.

[1] A. Vespignani, Modelling dynamical processes in complex socio-technical systems, Nat. Phys. **8**, 32 (2012).

[2] R. Pastor-Satorras, C. Castellano, P. Van Mieghem, and A. Vespignani, Epidemic processes in complex networks, Rev. Mod. Phys. **87**, 925 (2015).

[3] M. J. Keeling and K. T. D. Eames, Networks and epidemic models, J. R. Soc. Interface **2**, 295 (2005).

[4] G. Karlebach and R. Shamir, Modelling and analysis of gene regulatory networks, Nat. Rev. Mol. Cell Biol. **9**, 770 (2008).

[5] D.-U. Hwang, Complex networks: Structure and dynamics, Phys. Rep. **424**, 175 (2006).

[6] 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 **90**, 012801 (2014).

[7] F. Altarelli, A. Braunstein, L. Dall'Asta, A. Lage-Castellanos, and R. Zecchina, Bayesian inference of epidemics on networks via belief propagation, Phys. Rev. Lett. **112**, 118701 (2014).

[8] B. Aubin, B. Loureiro, A. Maillard, F. Krzakala, and L. Zdeborová, The spiked matrix model with generative priors, in *Advances in Neural Information Processing Systems* 32 (Curran Associates, Inc., 2019), pp. 8364–837.

[9] B. Aubin, B. Loureiro, A. Baker, F. Krzakala, and L. Zdeborová, Exact asymptotics for phase retrieval and compressed sensing with random generative priors, in *Proceedings of The First Mathematical and Scientific Machine Learning Conference, Proceedings of Machine Learning Research*, edited by J. Lu and R. Ward (PMLR, 2020), Vol. 107, pp. 55–73.

[10] M. Mezard and A. Montanari, *Information, Physics, and Computation* (Oxford University Press, New York, 2009).

[11] L. Zdeborová and F. Krzakala, Statistical physics of inference: Thresholds and algorithms, Adv. Phys. **65**, 453 (2016).

[12] J. Barbier, F. Krzakala, N. Macris, L. Miolane, and L. Zdeborová, Optimal errors and phase transitions in high-dimensional generalized linear models, Proc. Natl. Acad. Sci. USA **116**, 5451 (2019).

[13] A. Manoel, F. Krzakala, M. Mézard, and L. Zdeborová, Multi-layer generalized linear estimation, in *2017 IEEE International Symposium on Information Theory (ISIT)* (IEEE, New York, 2017), pp. 2098–2102.

[14] O. Duranthon and L. Zdeborová, Neural-prior stochastic block model, Mach. Learn.: Sci. Technol. **4**, 035017 (2023).

[15] M. E. J. Newman, The structure and function of complex networks, SIAM Rev. **45**, 167 (2003).

[16] I. Z. Kiss, J. C. Miller, P. L. Simon, *Mathematics of Epidemics on Networks From Exact to Approximate Models*, Interdisciplinary Applied Mathematics Vol. 46 (Springer, Cham, 2017).

[17] W. O. Kermack, A. G. McKendrick, and G. T. Walker, Contributions to the mathematical theory of epidemics. II. The problem of endemicity, Proc. R. Soc. London, Ser. A **138**, 55 (1932).

[18] P. E. Paré, C. L. Beck, and T. Başar, Modeling, estimation, and analysis of epidemics over networks: An overview, Annu. Rev. Control **50**, 345 (2020).

[19] D. Shah and T. Zaman, Detecting sources of computer viruses in networks: Theory and experiment, ACM SIGMETRICS Perform. Eval. Rev. **38**, 203 (2010).

[20] D. Shah and T. Zaman, Rumors in a network: Who's the culprit? IEEE Trans. Inf. Theor. **57**, 5163 (2011).

[21] N. Antulov-Fantulin, A. Lancic, H. Stefancic, M. Sikic, and T. Smuc, Statistical inference framework for source detection of contagion processes on arbitrary network structures, in *Proceedings of the 2014 IEEE Eighth International Conference on Self-Adaptive and Self-Organizing Systems Workshops* (IEEE, 2014), pp. 78–83.

[22] W. Dong, W. Zhang, and C. W. Tan, Rooting out the rumor culprit from suspects, in *Proceedings of the 2013 IEEE International Symposium on Information Theory* (IEEE, New York, 2013), pp. 2671–2675.

[23] P. C. Pinto, P. Thiran, and M. Vetterli, Locating the source of diffusion in large-scale networks, Phys. Rev. Lett. **109**, 068702 (2012).

[24] B. Karrer and M. E. J. Newman, Message passing approach for general epidemic models, Phys. Rev. E **82**, 016101 (2010).

[25] F. Altarelli, A. Braunstein, L. Dall'Asta, A. Ingrosso, and R. Zecchina, The patient-zero problem with noisy observations, J. Stat. Mech. (2014) P10016.

[26] A. Y. Lokhov, Reconstructing parameters of spreading models from partial observations, in *Advances in Neural Information Processing Systems* 29 (Curran Associates, Inc., 2016), pp. 3459–3467.

[27] A. Baker, I. Biazzo, A. Braunstein, G. Catania, L. Dall'Asta, A. Ingrosso, F. Krzakala, F. Mazza, M. Mézard, A. P. Muntoni, *et al.*, Epidemic mitigation by statistical inference from contact tracing data, Proc. Natl. Acad. Sci. USA **118**, e2106548118 (2021).

[28] M. Gabrié, A. Manoel, C. Luneau, J. Barbier, N. Macris, F. Krzakala, and L. Zdeborová, Entropy and mutual information in models of deep neural networks, J. Stat. Mech. (2019) 124014.

[29] E. Mallard and D. Saad, Inference by belief propagation in composite systems, Phys. Rev. E **78**, 021107 (2008).

[30] J. Raymond and D. Saad, Equilibrium properties of disordered spin models with two-scale interactions, Phys. Rev. E **80**, 031138 (2009).

[31] A. Braunstein, L. Budzynski, and M. Mariani, Statistical mechanics of inference in epidemic spreading, Phys. Rev. E **108**, 064302 (2023).

[32] A. Braunstein, L. Budzynski, and M. Mariani, Evidence of replica symmetry breaking under the Nishimori conditions in epidemic inference on graphs, Phys. Rev. E **111**, 064308 (2025).

[33] D. Ghio, Antoine L. M. Aragon, I. Biazzo, and L. Zdeborová, Bayes-optimal inference for spreading processes on random networks, Phys. Rev. E **108**, 044308 (2023).

[34] W. Luo, W. P. Tay, and M. Leng, Identifying infection sources and regions in large networks, IEEE Trans. Signal Process. **61**, 2850 (2013).

[35] K. Fan, A. E Aiello, and K. A Heller, Bayesian models for heterogeneous personalized health data, arXiv:1509.00110.

[36] E. Gardner, The space of interactions in neural network models, J. Phys. A: Math. Gen. **21**, 257 (1988).

[37] E. Gardner and B. Derrida, Optimal storage properties of neural network models, J. Phys. A: Math. Gen. **21**, 271 (1988).

[38] W. Krauth and M. Mézard, Storage capacity of memory networks with binary couplings, J. Phys. France **50**, 3057 (1989).

[39] B. Spinelli, L. E. Celis, and P. Thiran, A general framework for sensor placement in source localization, IEEE Trans. Network Sci. Eng. **6**, 86 (2017).

[40] H. Nishimori, *Statistical Physics of Spin Glasses and Information Processing: An Introduction* (Oxford University Press, Oxford, 2001).

[41] O. Duranthon and L. Zdeborová, Optimal inference in contextual stochastic block models, arXiv:2306.07948.

[42] J. S. Yedidia, W. T. Freeman, and Y. Weiss, Understanding belief propagation and its generalizations, in *Exploring Artificial Intelligence in the New Millennium*, edited by G. Lakemeyer and B. Nebel (Morgan Kaufmann, San Francisco, 2003), pp. 239–260.

[43] A. Coja-Oghlan, F. Krzakala, W. Perkins, and L. Zdeborová, Information-theoretic thresholds from the cavity method, Adv. Math. **333**, 694 (2018).

[44] M. Celentano, A. Montanari, and Y. Wu, The estimation error of general first order methods, in *Proceedings of the 33rd Conference on Learning Theory, Proceedings of Machine Learning Research* (PMLR, 2020), Vol. 125, pp. 1078–1141.

[45] A. Montanari and A. S Wein, Equivalence of approximate message passing and low-degree polynomials in rank-one matrix estimation, Probab. Theory Relat. Fields (2024).

[46] E. Cornacchia, F. Mignacco, R. Veiga, C. Gerbelot, B. Loureiro, and L. Zdeborová, Learning curves for the multi-class teacher–student perceptron, Mach. Learn.: Sci. Technol. **4**, 015019 (2023).

[47] D. Ghio, F. Boncoraglio, and L. Zdeborová, GitHub, 2025, https://github.com/IdePHICS/NeuralSpreadingInference.

[48] B. Bollobás, Random graphs, *Modern Graph Theory* (Springer, New York, 1998), pp. 215–252.