Belief propagation for permutations, rankings, and partial orders

George T. Cantwell and Cristopher Moore

Phys. Rev. E **105**, L052303 — Published 25 May 2022

DOI: 10.1103/PhysRevE.105.L052303

Belief propagation for permutations, rankings, and partial orders

George T. Cantwell and Cristopher Moore
Santa Fe Institute, 1399 Hyde Park Road, Santa Fe, New Mexico 87501, USA

Many datasets give partial information about an ordering or ranking by indicating which team won a game, which item a user prefers, or who infected whom. We define a continuous spin system whose Gibbs distribution is the posterior distribution on permutations, given a probabilistic model of these interactions. Using the cavity method we derive a belief propagation algorithm that computes the marginal distribution of each node's position. In addition, the Bethe free energy lets us approximate the number of linear extensions of a partial order and perform model selection between competing probabilistic models, such as the Bradley-Terry-Luce model of noisy comparisons and its cousins.

Ranking or ordering objects is a natural problem in many contexts. Mathematically, this task corresponds to finding "good" permutations of a finite set, or more generally, sampling from a distribution of good permutations. This can be surprisingly difficult.

For example, suppose we observe a set of pairwise interactions, such as competitions, preferences, or conflicts, each of which is evidence that one object is ranked above another, and our goal is to rank them from strongest to weakest. Similarly, we might want to reconstruct the order in which nodes joined a growing network [1, 2], for instance in an epidemic where contact tracing suggests links where one individual infected another. In cases like these, finding a permutation which minimizes the number of violations where the ordering goes the "wrong" way is NP-hard, i.e., among the hardest optimization problems in computer science [3]. Even when there exist permutations consistent with all observed interactions, counting the number of such permutations, or computing the average positionof a given object, is #P-complete [4, 5]. Thus all these problems are believed to take exponential time in the worst case.

Pairwise comparisons can be represented as a directed graph G whose edges (i, j) indicate that $i \prec j$, i.e., i "beat" j and is therefore probably ranked above j. We assume a generative model: given a ground-truth permutation $\boldsymbol{\pi}$, we observe G with probability $P(G|\boldsymbol{\pi})$ [6]. If all permutations are equally likely a priori, and if we observe each $i \prec j$ independently with probability $f(\pi_i, \pi_j)$, the posterior has the form

$$P(\boldsymbol{\pi}|G) = \frac{\prod_{(i,j)\in G} f(\pi_i, \pi_j)}{\sum_{\boldsymbol{\pi}'} \prod_{(i,j)\in G} f(\pi'_i, \pi'_j)}. \tag{1}$$

The framework of Eq. (1) may seem restrictive but we will see that it covers several interesting problems. Specifically, we consider (1) counting linear extensions of partial orders; (2) inferring the order in which a network grew; (3) finding minimum feedback arc sets; (4) parameter estimation and model selection for rankings.

To advance an analogy with statistical physics, we interpret Eq. (1) as a Gibbs distribution $P(\boldsymbol{\pi}|G) = e^{-\beta H(\boldsymbol{\pi})}/Z$ at temperature β^{-1} with Hamiltonian $H(\boldsymbol{\pi}) = \sum_{(i,j)\in G} h(\pi_i, \pi_j)$. Of particular interest is the step function Hamiltonian

$$h(\pi_i, \pi_j) = \Theta(\pi_i - \pi_j) = \begin{cases} 1 & \pi_i \geq \pi_j \\ 0 & \pi_i < \pi_j \,. \end{cases} \tag{2}$$

In this case, the energy $H(\boldsymbol{\pi})$ is the number of violations, i.e., the number of edges in G that are oriented contrary to the ordering of the nodes in permutation $\boldsymbol{\pi}$ [7, 8].

We begin our investigation of the Hamiltonian (2) at zero temperature. If the system is not frustrated, i.e. if there is a $\boldsymbol{\pi}$ for which $H(\boldsymbol{\pi}) = 0$, then we can view each directed edge $(i, j) \in G$ as a hard constraint that demands $\pi_i < \pi_j$. This implies that G is acyclic and defines a partial order: that is, a structure where we are given that $i \prec j$ for some pairs i, j. Partial orders are transitive: $i \prec j$ and $j \prec k$ implies $i \prec k$. However, there may be pairs i, j where neither $i \prec j$ nor $j \prec i$ is necessarily true, leaving their relative order ambiguous. In contrast, a *total* or *linear* order is a permutation—every object has an unambiguous rank, and all pairs of items are comparable. A *linear extension* of a partial order is a total order that satisfies all the constraints of the partial order; it is is equivalent to a topological ordering of the corresponding directed acyclic graph. There are typically many such orders.

Counting the linear extensions of a partial order is a well-known problem in computer science. Counting them exactly is #P-complete [5], making it as hard as computing spin glass partition functions or matrix permanents, and almost certainly requiring exponential time. There are polynomial-time Monte Carlo algorithms [9–11] but these are fairly slow in practice [12]. Here we provide a fast approximate algorithm based on belief propagation and the cavity method in a related spin system.

Permutation-valued states cause several challenges for the cavity method. First, *a priori* the entropy grows super-extensively as $\log n! \sim n \log n$, creating a rather odd thermodynamic limit. Secondly, since each site can be in one of n different states, we have something like a q-state Potts model where $q = n$, making the fields n-dimensional. Thirdly, no two sites can have the same state. This creates a global coupling, violating the local treelikeness that the cavity method assumes.

We can address all these problems with a rescaling to

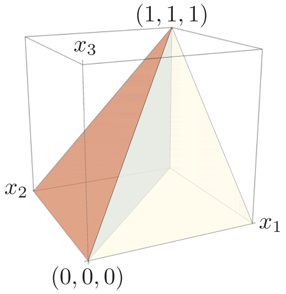

FIG. 1. The convex polytope corresponding to the partial order $3 \prec 1, 3 \prec 2$, i.e., the subset of the unit cube where $x_3 < x_1$ and $x_3 < x_2$. It contains two simplices corresponding to the linear extensions $3 \prec 1 \prec 2$ and $3 \prec 2 \prec 1$, and has total volume $2/3! = 1/3$.

continuous variables. Rather than treat each π_i as an integer from 1 to n, we associate it with a real number x_i in the unit interval. A state is a point in the n-dimensional unit hypercube, $\boldsymbol{x} \in [0,1]^n$, which corresponds to the permutation given by the sorted order of its components. Since the x_i are distinct with probability 1, this removes the global coupling between sites.

Under this rescaling, the set of linear extensions of a given partial order becomes a convex polytope whose facets correspond to its constraints. Figure 1 shows the polytope corresponding to the partial order $3 \prec 1, 3 \prec 2$, which has facets $x_3 < x_1$ and $x_3 < x_2$. This partial order has two linear extensions, $3 \prec 1 \prec 2$ and $3 \prec 2 \prec 1$.

Although our variables x_i live in the unit interval as opposed to the circle or sphere, we think of this rescaled model as a continuous spin system in the spirit of the XY model or the classical Heisenberg model [13]. At zero temperature the partition function Z is the volume of the polytope of linear extensions. Since each permutation corresponds to a simplex with volume $1/n!$, this gives

$$\begin{aligned} Z &= \lim_{\beta\to\infty} \int_{[0,1]^n} \prod_{(i,j)\in G} e^{-\beta\Theta(x_i - x_j)}\,\mathrm{d}\boldsymbol{x} \\ &= \frac{\#\text{ linear extensions}}{n!}\,, \end{aligned} \tag{3}$$

and because every linear extension is equally likely, the entropy is simply $S = \ln Z$. As we discuss below, this rescaling from $\{1, \ldots, n\}$ to the unit interval allows us to define a sensible thermodynamic limit where Z behaves as a simple exponential and S is linear in n.

Since counting linear extensions is #P-complete, so is computing Z or S exactly. We will use the cavity method to approximate them. We start by pretending that the graph of comparisons is a tree—that it has no cycles even when the directions of the edges are erased. The distribution of the spins can then be factorized as

$$P(\boldsymbol{x}|G) = \frac{\prod_{(i,j)\in G} \mu_{ij}(x_i, x_j)}{\prod_i \mu_i(x_i)^{d_i - 1}}\,. \tag{4}$$

Here $\mu_i(x_i)$ is the marginal probability density for the spin at node i, $\mu_{ij}(x_i, x_j)$ is the joint marginal for the spins at nodes i and j, and d_i is the degree of node i, i.e., the number of objects to which it is compared. The entropy $S = -\langle \ln P \rangle$ is then given by the *Bethe entropy*

$$S_{\text{Bethe}} = \sum_{(i,j)\in G} S_{ij} - \sum_i (d_i - 1) S_i\,, \tag{5}$$

where $S_i = -\big\langle \ln \mu_i \big\rangle$ and $S_{ij} = -\big\langle \ln \mu_{ij} \big\rangle$ are the entropies of the one- and two-point marginals respectively.

These marginal distributions can be computed using belief propagation [14, 15]. For each neighboring pair (i, j), we ask what j's marginal would be if i were absent. We denote this cavity marginal $\mu_{j\to i}(x)$, and think of it as a "message" or "belief" that j sends to i. It in turn depends on the messages that j receives from its neighbors k other than i. Note that if i and j are compared, messages go in both directions along the edge (i, j).

For linear extensions, $\mu_{j\to i}(x_j)$ is proportional to the probability that $x_k < x_j$ (resp. $x_k > x_j$) for all $k \neq i$ such that $k \prec j$ (resp. $k \succ j$). Using the cumulative distribution functions

$$M_{j\to i}(x_j) = \int_0^{x_j} \mu_{j\to i}(y)\,\mathrm{d}y\,, \tag{6}$$

we can write this as

$$\mu_{j\to i}(x_j) \propto \prod_{\substack{k\prec j \\ k\neq i}} M_{k\to j}(x_j) \times \prod_{\substack{k\succ j \\ k\neq i}} \big(1 - M_{k\to j}(x_j)\big)\,, \tag{7}$$

where we normalize so that $\int_0^1 \mu_{j\to i}(x_j)\,\mathrm{d}x_j = 1$. The one-point marginals are computed similarly, but using all of j's neighbors,

$$\mu_j(x_j) \propto \prod_{k\prec j} M_{k\to j}(x_j) \times \prod_{k\succ j} \big(1 - M_{k\to j}(x_j)\big)\,, \tag{8}$$

and the two-point marginal for an edge $i \prec j$ is

$$\mu_{ij}(x_i, x_j) \propto \mu_{j\to i}(x_j)\,\mu_{i\to j}(x_i)\,\Theta(x_j - x_i)\,. \tag{9}$$

Note that we assume a uniform prior on the hypercube, and therefore a uniform prior on permutations.

This suggests an algorithm for counting linear extensions: solve the belief propagation equations (7) by iterating until we reach a fixed point, and compute the entropy $S = \ln Z$ in Eq. (5). If the graph of comparisons has loops, this algorithm is not exact, but the Bethe entropy is often an excellent asymptotic approximation to the true entropy. In particular, so long as the graph is sparse and locally treelike (with few short loops) we expect the resulting estimate of Z to be correct up to subexponential terms.

Writing the system of equations in Eq. (7) is one thing but solving it another, since it consists of a large system

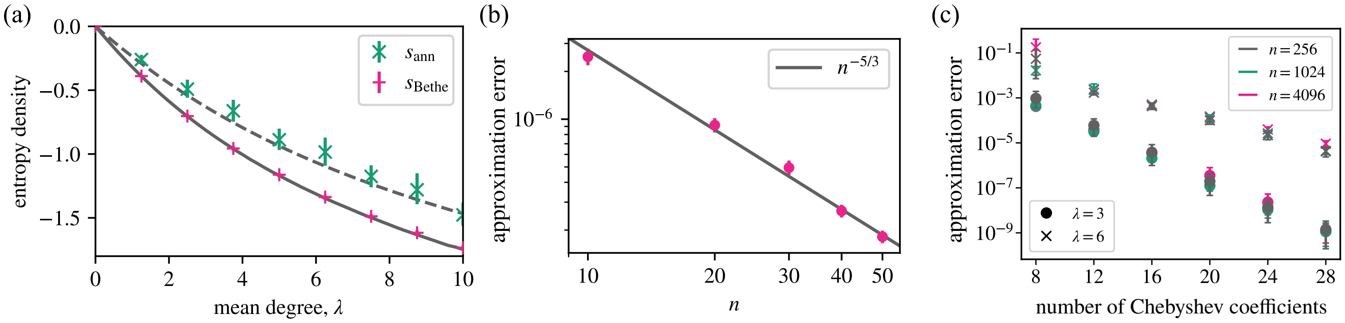

FIG. 2. Entropy per site for random graphs. The number of linear extensions scales as $n!e^{sn}$ in the sparse case where $s < 0$, so after rescaling the spins to the unit interval as in Eq. (3) the entropy is negative. (a) Values of $s_{\rm ann}$ and $s_{\rm Bethe}$ in random graphs of mean degree λ, estimated by belief propagation on random graphs of size $n = 10^4$ under the assumption that $\ln Z$ is normally distributed. The dashed line is the analytic result Eq. (10) for $s_{\rm ann}$, showing that our results are consistent with theory. The solid line is the value for $s_{\rm Bethe}$ given by population dynamics. (b) Mean-squared error between $s_{\rm Bethe}$ and the exact value of $s = \frac{1}{n}\ln Z$ computed by exhaustive enumeration on random graphs of size $n \leq 50$ and mean degree $\lambda = 2$. Even when we approximate the messages with $d = 32$ Chebyshev coefficients, $s_{\rm Bethe}$ rapidly converges to s, showing that the cavity method is asymptotically exact. (c) Mean-squared error for our estimate of $s_{\rm Bethe}$ as a function of the number d of Chebyshev coefficients used in our approximation, for random graphs with n nodes and mean degree λ. The error is calculated relative to $d = 32$. There is clear dependence on λ but not on n, and the estimate converges exponentially as d increases.

of nonlinear differential equations. Some further insights, however, reduce the complexity considerably. First, note that for any partial order the true marginal distribution $\mu_i(x)$ is a polynomial of degree at most $n-1$. To see this, recall that the unit hypercube can be divided into $n!$ simplices, each of which corresponds to one permutation. If i's position in this permutation is t, then $t-1$ of the other spins x_j must be less than x_i, and the other $n-t$ spins must be greater. The probability density for x_i is thus proportional to $x^{t-1}(1-x)^{n-t}$, a so-called Bernstein polynomial. Summing over all allowed permutations, $\mu_i(x)$ is a linear combination of such polynomials. Similarly, the messages $\mu_{j\to i}(x)$ are polynomials of degree at most $n-2$.

This would allow us to solve Eq. (7) by finding at most $n-1$ polynomial coefficients fonor the right-hand side. However, this would require $O(n)$ computation for each edge. Happily, this is unnecessary: we can approximate the messages as polynomials of lower degree using Chebyshev polynomials [16] $T_k(x)$, up to some maximum degree d, writing $\mu_{j\to i}(x) \approx \sum_{k=0}^{d-1} c_{j\to i}^k T_k(x)$. We initialize the messages to uniform distributions $T_0 = 1$, i.e., $\boldsymbol{c}_{j\to i} = (1, 0, \ldots, 0)$. We then iteratively update $\boldsymbol{c}_{j\to i}$ and $\boldsymbol{c}_{i\to j}$ for all edges (i, j) using Eq. (7).

Since the Chebyshev polynomial of degree k on $[0, 1]$ can be written $T_k(x) = \cos k\theta$ where $x = (1+\cos\theta)/2$, we can compute each update using the Fast Fourier Transform. On sparse graphs, where the degree distribution has finite mean and variance, the computation time for an entire sweep is linear in the number of edges, and we typically converge to a fixed point in $O(\log n)$ sweeps. In practice we obtain excellent results even when the polynomial degree d is considerably smaller than n (see Fig. 2), allowing our method to scale easily to hundreds of thousands of nodes on a desktop computer.

As an initial test of our methods, suppose G is a directed version of an Erdős-Rényi graph $G_{n,p}$ where each pair of nodes is compared with probability p. To create a valid partial order, we label the nodes $1, \ldots, n$ with a ground-truth permutation π, and orient the edges to agree with π. This model of a random partial order was studied in [17]. For sparse graphs with average degree λ, i.e., in the limit $n \to \infty$ and $p = \lambda/n$, by applying results in Ref. [17] we derive the annealed entropy per site

$$s_{\rm ann} = \lim_{n\to\infty} \frac{\ln\langle Z_n\rangle}{n} = 1 - \ln\lambda + \int_0^1 \ln(1 - e^{-\lambda x})\,dx\,, \quad (10)$$

so the expected number of linear extensions is $\langle n! Z_n\rangle = n! e^{n s_{\rm ann}}$ up to subexponential terms.

Note the unusual scaling of this problem. In sparse graphs where $\lambda = O(1)$ and there are $O(n)$ edges, each edge excludes a constant fraction of permutations, so the number of linear extensions is $n!$ multiplied by a simple exponential e^{sn} with $s < 0$. After rescaling to the unit interval as in (3), the total volume Z is e^{sn}, giving a valid thermodynamic limit where the entropy is extensive. In contrast, in dense graphs with $O(n^2)$ edges, almost all of the $n!$ possible permutations are excluded, and the number of linear extensions is a simple exponential [17].

We conjecture that the cavity method is asymptotically exact in sparse random graphs: namely, that the typical quenched entropy is given by the Bethe entropy, $\lim_{n\to\infty} \frac{1}{n}\langle \ln Z_n\rangle = \lim_{n\to\infty} \frac{1}{n} S_{\rm Bethe} = s_{\rm Bethe}(\lambda)$. To test this conjecture, and measure the quality of our Chebyshev approximation, we use population dynamics [15] to derive a fixed-point distribution of messages on sparse infinite graphs to estimate $s_{\rm Bethe}$, and also carried out belief propagation on finite graphs.

By running belief propagation on multiple realizations of finite random graphs, we estimated the mean μ and variance σ^2 of $\ln Z$. On the physical assumption that $\ln Z$ is normally distributed (which was proved for the dense case in [17] and is easy to prove for the sparse case when λ is sufficiently small) we have $s_{\text{Bethe}} = \mu/n$ and $s_{\text{ann}} = (\mu + \frac{1}{2}\sigma^2)/n$.

Figure 2(a) shows that the resulting estimates of s_{ann} reproduce the analytic result Eq. (10), showing that our method is consistent with theory. We also find that s_{Bethe} from belief propagation on finite graphs agrees closely with the prediction from population dynamics.

We compared s_{Bethe} with the exact entropy $s = \frac{1}{n}\ln(\#\text{ linear extensions}/n!)$, using a dynamic programming algorithm to enumerate permutations exhaustively, on random graphs of size up to $n = 50$. Fig. 2(b) shows that s_{Bethe} converges quickly to the true entropy as n increases, as $n^{-\alpha}$ where $\alpha \approx 5/3$, suggesting that the cavity method is asymptotically correct.

The results of Fig. 2(a,b) were obtained using Chebyshev polynomials of degree $d = 32$. While the exact marginals have degree $n-1$, we observe exponential convergence in the number d of coefficients used in our Chebyshev approximation. As Fig. 2(c) shows, if we fix the average degree λ of the graph, our estimate of s_{Bethe} converges rapidly as d increases, and the error does not depend on n. Thus our approximation of the messages with low-degree polynomials is also highly accurate.

Although counting linear extensions of a partial order is an established problem computer science, it is less familiar in physics. It is not so far removed, however, from inference in growing network models [1, 2] such as preferential attachment [18, 19], which are among the most widely studied models in the physics of complex networks. If G_t is the network at time t, these models specify a transition probability $P(G_{t+1}|G_t)$ along with an initial condition G_0. Since we often only observe a snapshot of the network, or its final state, an interesting problem is to reconstruct the history of the network, i.e., the order in which its nodes were added. For trees, one can calculate the full distribution of possible "histories" by which an observed graph could have grown [20]. Our methods here provide an approximate solution for general graphs.

If the network's edges are directed, pointing from each new node to the node it attached to, they constitute a partial order with which the network's history must be consistent. We can then use Eq. (8) to compute the posterior distribution of each node's arrival time. We show an example in Fig. 3. This graph is sufficiently small ($n = 10$) that we can compare our method with an exhaustive enumeration of all $n!$ possible orderings. As shown in panel (b), the marginals we obtain are very accurate despite the presence of short loops. Thus we can efficiently approximate, not just the mean arrival time of each node, but its posterior distribution.

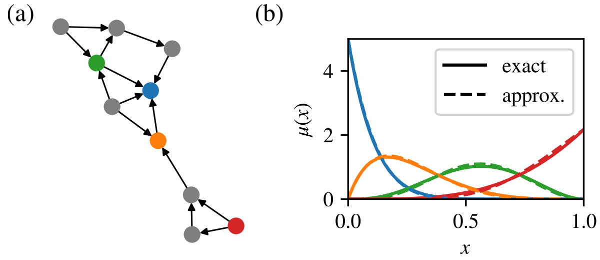

FIG. 3. In (a) we show a randomly grown network with Poisson out-degree. In (b) we show the posterior marginals for four representative nodes, colored to match (a), comparing those obtained by our method with the exact results of exhaustive enumeration. Despite the presence of short cycles, our belief propagation approach approximates the marginals quite closely, matching not just the means but the shapes of these distributions.

Next we turn to the frustrated case, where G contains directed cycles such as $1 \prec 2 \prec 3 \prec 1$. In this case, no permutation satisfies all the constraints and the data is inconsistent with $\beta = \infty$. At finite β, Eq. (7) becomes

$$\mu_{j\to i}(x) \propto \prod_{k\neq i} \int_0^1 \frac{\mu_{k\to j}(y)\, e^{-\beta g_{jk}\Theta(x-y)}}{e^{-\beta\Theta(x-y)} + e^{-\beta\Theta(y-x)}}\, \mathrm{d}y \tag{11}$$

$$= \prod_{k\neq i} \int_0^1 \frac{\mu_{k\to j}(y)\, e^{-\beta g_{jk}\Theta(x-y)}}{1 + e^{-\beta}}\, \mathrm{d}y \tag{12}$$

where $g_{jk} = 1$ if $k \prec j$ and -1 if $k \succ j$. The marginals are again polynomials of degree at most $n-1$ since the density is a function only of $\boldsymbol{x}$'s permutation; we calculate these integrals as before using Chebyshev polynomials.

The ground states of the Hamiltonian (2) are permutations that minimize the number of violated constraints. These are known as minimum violation rankings [7, 8]. Like counting linear extensions of a partial order, finding these rankings is computationally hard. It is equivalent to the classic NP-complete "minimum feedback arc set" problem of making a directed graph acyclic by removing as few edges as possible [3]. Intuitively, this is because the landscape of rankings can be glassy, with multiple widely separated peaks. For instance, given the comparisons $1 \prec 2 \prec \cdots \prec n$, along with $n \prec 1$ and $1 \prec (n-1)$, then there are two minimum violation orderings, one where n is first and another where it is last.

One strategy to find the frustrated ground states is to perform belief propagation at low temperature, where they dominate the Gibbs distribution. If we wish to find a single ground state regardless of its probability, we can use greedy decimation, removing the edge (i,j) most likely to be violated according to μ_{ij} and iterating. Belief propagation for this problem was previously studied, using a discrete model, in Ref. [21]. We ran both our procedures on 40 random directed graphs with $n = 1000$ and mean degree $\lambda = 8$ at $e^{-\beta} = 2^{-10}$, and compared with nine other scalable methods from computer science [22, 23]. Both belief propagation algorithms obtained the smallest number of violations; ours is somewhat faster, perhaps because we use a continuous model and low-degree polynomials.

We turn next to parameter estimation. If we do not know how noisy comparisons are, we should use the data to estimate β. The total probability that a model will generate G, averaged over all permutations $\boldsymbol{\pi}$, is

$$\begin{aligned} P(G) &= \frac{1}{n!} \sum_{\boldsymbol{\pi}} \prod_{(i,j)\in G} f(\pi_i, \pi_j) \\ &= \int_{[0,1]^n} \prod_{(i,j)\in G} \frac{e^{-\beta\Theta(x_i - x_j)}}{1+e^{-\beta}} = Z(\beta) \end{aligned} \tag{13}$$

Using the thermodynamic relation $Z = e^{S-U}$ we approximate $Z(\beta)$ from the Bethe entropy and the energy, where the cavity approximation for the energy is

$$U = -\sum_{(i,j)\in G} \int_0^1 \int_0^1 \mu_{ij}(x,y) \ln f(x,y)\, \mathrm{d}x\, \mathrm{d}y. \tag{14}$$

If we have no prior information about β, we can determine its most-likely value by maximizing $Z(\beta)$.

Finally, we discuss model selection. How can we choose between two different model classes, i.e., two different Hamiltonians? We might want to choose between the step-function model, where the probability that $i \prec j$ or $i \succ j$ depends only on the relative order of these two items, and a model where these probabilities depend on how far apart they are, for example the Bradley-Terry-Luce (BTL) model [24] (which actually dates back to Zermelo [25]), or SpringRank [26]. Let us consider BTL—a popular model of user preferences, similar to Elo Chess ratings where the probability a user prefers i to j or that i will beat j in a chess game is a logistic function

$$f(x_i, x_j) = \frac{e^{\beta(x_i - x_j)}}{e^{\beta(x_i-x_j)} + e^{\beta(x_j - x_i)}} = \frac{r_i}{r_i + r_j} \tag{15}$$

where $r_i = e^{2\beta x_i}$. In general the continuous ranks in this model are allowed to range over the real line. We can scale them to the unit interval, and thus use our Chebyshev approximation, by varying β. This corresponds to assuming a uniform prior of width β on the x_i. To analyze the model we simply replace $\Theta(x-y)$ with $x-y$ in the belief propagation equations, Eq. (11). We can use the orthogonality of the Chebyshev polynomials to efficiently compute the integral as a matrix product.

Given the observed comparisons, which of these two models should we prefer? Both the models have one free parameter, and so a simple approach is to prefer the model with the largest maximum likelihood.

Using our methods, we computed the maximum likelihood values for ATP tennis tournament matches for the ten years 2010–2019 [27]. On average there were $n = 435$ players. The average degree, i.e., the average number of games each player played it, was number of interactions $\lambda = 11.0$. We found, interestingly, that the BTL model is only preferred over the step function model in one of the ten years (2010). In other years, the data is better explained by the step function model—where the probability that a weaker player beats a stronger one is fixed, rather than depending on the difference in their ranks.

To summarize, we have shown how natural problems involving permutations, rankings, and orderings can be treated as continuous spin systems. This includes counting linear extensions of a partial order, inferring the order in which nodes joined a growing network, and finding minimum-violation rankings. We discussed both models where probabilities depend only on the ordering, and those such as the Bradley-Terry-Luce model where they depend on differences in rank. We derived an efficient belief propagation algorithm using low-degree polynomials to compute marginals and entropies, and found that it is accurate on both sparse random graphs and some graphs with short loops. By using the Bethe free energy as an estimate of the log-likelihood, it can also perform parameter estimation and model comparison, and can be readily applied to real-world data.

Acknowledgments. This work was supported by NSF grant BIGDATA-1838251. We thank Jiaming Xu and Jean-Gabriel Young for helpful conversations. Code implementing our methods is available at https://github.com/gcant/pairwise-comparison-BP.

[1] J.-G. Young, G. St-Onge, E. Laurence, C. Murphy, L. Hébert-Dufresne, and P. Desrosiers, Phase transition in the recoverability of network history. *Physical Review X* **9**(4), 041056 (2019), URL https://link.aps.org/doi/10.1103/PhysRevX.9.041056.

[2] S. Navlakha and C. Kingsford, Network archaeology: Uncovering ancient networks from present-day interactions. *PLOS Computational Biology* **7**(4), 1–16 (2011), URL https://doi.org/10.1371/journal.pcbi.1001119.

[3] R. M. Karp, Reducibility among combinatorial problems. In R. E. Miller, J. W. Thatcher, and J. D. Bohlinger (eds.), *Complexity of Computer Computations*, pp. 85–103, Springer US, Boston, MA (1972), URL http://link.springer.com/10.1007/978-1-4684-2001-2_9.

[4] G. Brightwell and P. Winkler, Counting linear extensions is #P-complete. In *Proceedings of the twenty-third annual ACM symposium on Theory of computing - STOC '91*, pp. 175–181, ACM Press (1991), URL http://portal.acm.org/citation.cfm?doid=103418.103441.

[5] G. Brightwell and P. Winkler, Counting linear extensions. *Order* **8**(3), 225–242 (1991), URL http://link.springer.com/10.1007/BF00383444.

[6] Our model does not attempt to explain which pairs interact, only the outcome of these interactions.

[7] W. A. Thompson and R. Remage, Rankings from paired comparisons. *The Annals of Mathematical Statis-*

tics **35**(2), 739–747 (1964), URL http://www.jstor.org/stable/2238526.

[8] J. Park, Diagrammatic perturbation methods in networks and sports ranking combinatorics. *Journal of Statistical Mechanics: Theory and Experiment* **2010**(04), P04006 (2010), URL https://iopscience.iop.org/article/10.1088/1742-5468/2010/04/P04006.

[9] M. Dyer, A. Frieze, and R. Kannan, A random polynomial-time algorithm for approximating the volume of convex bodies. *Journal of the ACM* **38**(1), 1–17 (1991), URL https://dl.acm.org/doi/10.1145/102782.102783.

[10] A. Karzanov and L. Khachiyan, On the conductance of order Markov chains. *Order* **8**(1), 7–15 (1991), URL http://link.springer.com/10.1007/BF00385809.

[11] L. Lovász and M. Simonovits, Random walks in a convex body and an improved volume algorithm. *Random Structures and Algorithms* **4**(4), 359–412 (1993), URL https://onlinelibrary.wiley.com/doi/10.1002/rsa.3240040402.

[12] T. Talvitie, K. Kangas, T. Niinimäki, and M. Koivisto, Counting linear extensions in practice: MCMC versus exponential Monte Carlo. In *AAAI Conference on Artificial Intelligence* (2018), URL https://aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16957.

[13] H. E. Stanley, Dependence of critical properties on dimensionality of spins. *Phys. Rev. Lett.* **20**, 589–592 (1968), URL https://link.aps.org/doi/10.1103/PhysRevLett.20.589.

[14] J. Pearl, Reverend Bayes on inference engines: A distributed hierarchical approach. In *Proc. 2nd AAAI Conference on Artificial Intelligence*, pp. 133–136 (1982).

[15] M. Mézard and A. Montanari, *Information, physics, and computation*. Oxford graduate texts, Oxford University Press, Oxford (2009).

[16] L. N. Trefethen, *Approximation Theory and Approximation Practice, Extended Edition*. SIAM (2019).

[17] N. Alon, B. Bollobás, G. Brightwell, and S. Janson, Linear extensions of a random partial order. *The Annals of Applied Probability* **4**(1) (1994), URL https://doi.org/10.1214/aoap/1177005202.

[18] D. J. de Solla Price, Networks of scientific papers. *Science* **149**(3683), 510–515 (1965), URL https://www.sciencemag.org/lookup/doi/10.1126/science.149.3683.510.

[19] R. Albert and A.-L. Barabási, Statistical mechanics of complex networks. *Reviews of Modern Physics* **74**(1), 47–97 (2002), URL https://link.aps.org/doi/10.1103/RevModPhys.74.47.

[20] G. T. Cantwell, G. St-Onge, and J.-G. Young, Inference, model selection, and the combinatorics of growing trees. *Physical Review Letters* **126**(3), 038301 (2021), URL https://link.aps.org/doi/10.1103/PhysRevLett.126.038301.

[21] J.-H. Zhao and H.-J. Zhou, Feedback arcs and node hierarchy in directed networks. *Chinese Physics B* **26**(7), 078901 (2017), URL https://doi.org/10.1088/1674-1056/26/7/078901.

[22] P. Festa, P. M. Pardalos, and M. G. C. Resende, Algorithm 815: FORTRAN subroutines for computing approximate solutions of feedback set problems using GRASP. *ACM Transactions on Mathematical Software* **27**(4), 456–464 (2001), URL https://dl.acm.org/doi/10.1145/504210.504214.

[23] M. Simpson, V. Srinivasan, and A. Thomo, Efficient computation of feedback arc set at web-scale. *Proceedings of the VLDB Endowment* **10**(3), 133–144 (2016), URL https://dl.acm.org/doi/10.14778/3021924.3021930.

[24] R. A. Bradley and M. E. Terry, Rank analysis of incomplete block designs: I. The method of paired comparisons. *Biometrika* **39**(3/4), 324 (1952), URL https://www.jstor.org/stable/2334029?origin=crossref.

[25] E. Zermelo, Die Berechnung der Turnier-Ergebnisse als ein Maximumproblem der Wahrscheinlichkeitsrechnung. *Mathematische Zeitschrift* **29**(1), 436–460 (1929), URL http://link.springer.com/10.1007/BF01180541.

[26] C. De Bacco, D. B. Larremore, and C. Moore, A physical model for efficient ranking in networks. *Science Advances* **4**(7), eaar8260 (2018), URL https://advances.sciencemag.org/lookup/doi/10.1126/sciadv.aar8260.

[27] Jeff Sackmann, ATP Tennis Rankings, Results, and Stats. URL https://github.com/JeffSackmann/tennis_atp.