Inference and optimization of real edges on sparse graphs: A statistical physics perspective

K. Y. Michael Wong[1] and David Saad[2]

[1]*Department of Physics, The Hong Kong University of Science and Technology, Hong Kong, China*
[2]*Neural Computing Research Group, Aston University, Birmingham B4 7ET, United Kingdom*

Inference and optimization of real-value edge variables in sparse graphs are studied using the Bethe approximation and replica method of statistical physics. Equilibrium states of general energy functions involving a large set of *real* edge variables that interact at the network nodes are obtained in various cases. When applied to the representative problem of network resource allocation, efficient distributed algorithms are also devised. Scaling properties with respect to the network connectivity and the resource availability are found, and links to probabilistic Bayesian approximation methods are established. Different cost measures are considered and algorithmic solutions in the various cases are devised and examined numerically. Simulation results are in full agreement with the theory.

I. INTRODUCTION

The links between statistical physics models and a variety of inference and optimization problems have been significantly strengthened over the last decade [1]. Two aspects of these links have been exploited. Macroscopically, using the statistical physics framework, one describes typical properties of the problem and provides valuable insight into its generic characteristics. Microscopically, established techniques of statistical physics such as the cavity method have been used for devising efficient inference algorithms, some of which have been independently discovered and used in other research communities [2–5].

Most studies so far, both within and outside the statistical physics community, have focused on cases of discrete variables. Among the recently successful examples using methods of statistics-based mechanics, one can mention hard computational problems [6] and error-correcting codes [7]. Statistical mechanical approaches to learning of discrete variables have also been considered on tree structures [8].

On the other hand, networks of continuous variables were much less explored. One of the main reasons for this limited activity is the difficulty in applying message passing approximation algorithms [2,3] in this case, as the discrete messages passed between variables become *functions* of real variables. Applied message passing for systems of real variables typically relies on modeling the functions using a reduced number of parameters [9].

In the statistical physics community there have been recent attempts to simplify the messages for continuous variables. For example, a step forward was made in Ref. [10] to parametrize the messages using eigenfunction decomposition for special cases. Furthermore, the continuous variables treated by these methods are *localized* on nodes, whereas many interesting problems, such as the resource allocation problem presented here (and partially in [11,12]), involve real variables defined on links *between* nodes.

In this paper we study a system with real variables that can be mapped onto a sparse graph and suggest an efficient message-passing approximation method for inference and optimization. We first formulate the problem at a general

temperature; the message-passing algorithm we present here as well as the related analysis are primarily *general* inference algorithms. In this paper, however, we are particularly interested in the optimal, zero temperature solution that reduces the task to an optimization problem.

Global optimization techniques, such as linear or quadratic programming [13], can successfully solve many of these problems. However, message-passing approaches have the potential to solve global optimization problems via *local* updates, thereby reducing the growth in computational complexity from cubic to linear with the system size. An even more important practical advantage is its distributive nature that is particularly suitable for distributive computation in large or evolving networks and does not require a global optimizer.

We focus on a prototype for optimization, and use the example of resource allocation as a vehicle to demonstrate the potential of our method, both for gaining insights into the main properties of the system and as an efficient optimization algorithm. Our method is efficient since the messages consist of only the first and second derivatives of the vertex free energies derived from our analysis. The key to the successful simplification, not needed for the simpler case of discrete variables, is that the messages passed to a target node are accompanied by information-provision messages from the target node, to first determine the working point at which the derivatives should be calculated.

The problem of resource allocation is a well-known network problem in the areas of computer science and operations management [14,15]. The problem itself is quite general and is applicable to typical situations where a large number of nodes are required to balance loads and/or resources, such as reducing Internet traffic congestion and streamlining network flow of commodities [16,17]. In computer science, many practical algorithmic solutions have been proposed to distribute the computational load between computers connected in a network. They usually are heuristic and focus on practical aspects (e.g., communication protocols). The problem we are addressing here is more generic and, in the context of computer networks, is represented by nodes of some computational power that should carry out tasks; subtasks are then moved around such that all demands

will be satisfied while the migration of (sub-)tasks is minimized.

In Sec. II we will introduce the general model, followed by a replica-based analysis in Sec. III and subsequently by a Bethe approximation-based analysis in Sec. IV. A message-passing algorithm for the problem of resource allocation will be presented in Sec. V followed by the derivation of scaling laws in the limit of high connectivity in Sec. VI. Numerical results for Secs. III–VI will be presented in Sec. VII. We will then extend the model to the case of general cost functions in Sec. VIII, highlighting strengths and weaknesses of our approach. We will conclude the presentation with a summary and point to future research directions.

II. MODEL

The problem we are addressing here is a generic version of resource allocation and serves as an example of a sparsely connected system of real variables that should be optimized with respect to some general cost. It is represented by nodes of some computational power that should carry out tasks. Both computational powers and tasks will be chosen at random from some arbitrary distribution. The nodes are located on a randomly chosen sparse network of some connectivity. The goal is to migrate tasks on the network such that demands will be satisfied while the migration of (sub-)tasks is minimized. We focus here on the satisfiable case where the total computing power is greater than the demand, and where the number of nodes involved is very large.

The sparse network considered has N nodes, labeled $i = 1, \ldots, N$. Each node i is randomly connected to c other nodes. The connectivity matrix is given by $\mathcal{A}_{ij} = \mathcal{A}_{ji} = 1, 0$ for connected and unconnected node pairs, respectively. A link variable y_{ij} is defined on each connected link from j to i. We focus on the case of intensive connectivity $c \sim O(1) \ll N$; and restrict the problem to the fixed connectivity case although both the analysis and the algorithm devised on its basis can handle a general connectivity profile.

We consider a general energy function (cost)

$$E = \sum_{(ij)} \mathcal{A}_{ij} \phi(y_{ij}) + \sum_i \psi(\Lambda_i, \{y_{ij} | \mathcal{A}_{ij} = 1\}),$$

where the summation (ij) is made over all node pairs, and Λ_i is a quenched variable defined on node i. In the context of probabilistic inference, y_{ij} may represent the coupling between observables in nodes j and i, $\phi(y_{ij})$ may correspond to the logarithm of the prior distribution of y_{ij}, and $\psi(\Lambda_i, \{y_{ij} | \mathcal{A}_{ij} = 1\})$ the logarithm of the likelihood of the observables Λ_i. Since the cost is independent of the direction of the currents in many applications, we focus on the case that $\phi(y)$ is a general even function of y. In the context of resource allocation, $y_{ij} \equiv -y_{ji}$ may represent the current from node j to i, $\phi(y_{ij})$ may correspond to the transportation cost, and $\psi(\Lambda_i, \{y_{ij} | \mathcal{A}_{ij} = 1\})$ the performance cost of the allocation task on node i, dependent on the node capacity Λ_i; the capacity of a node is defined as its computational capability minus its computational demand, and is randomly drawn from a distribution $\rho(\Lambda_i)$.

III. REPLICA ANALYSIS

To make the analysis more concrete and strengthen the link to the resource allocation problem, we keep the term $\phi(y_{ij})$ general and, aiming to satisfy the capacity constraints, set $\psi(\Lambda_i, \{y_{ij} | \mathcal{A}_{ij} = 1\}) = \ln[\Theta(-\Sigma_j \mathcal{A}_{ij} y_{ij} - \Lambda_i) + \epsilon]$, where $\epsilon \to 0$ and Θ is the step function. This reduces the problem to the load balancing task of minimizing the energy function (cost) $E = \Sigma_{(ij)} \mathcal{A}_{ij} \phi(y_{ij})$, subject to the constraints on the resources of nodes i,

$$\sum_j \mathcal{A}_{ij} y_{ij} + \Lambda_i \geq 0. \tag{1}$$

An alternative formulation is to consider the dual of the original optimization problem. Introducing Lagrange multipliers, the function to be minimized becomes

$$L = \sum_{(ij)} \mathcal{A}_{ij} \phi(y_{ij}) + \sum_i \mu_i \left(\sum_j \mathcal{A}_{ij} y_{ij} + \Lambda_i \right). \tag{2}$$

Optimizing L with respect to y_{ij}, one obtains

$$y_{ij} = [\phi']^{-1}(\mu_j - \mu_i), \tag{3}$$

where μ_i is referred to as the *chemical potential* of node i, and ϕ' is the derivative of ϕ with respect to its argument. This can be interpreted as the current being driven by the potential difference.

Since the probability of finding loops of finite lengths is vanishing in large sparse networks, the structure of a sparse network is locally a tree. Thus, given a configuration of currents $\{y_{ij}\}$, one can set the *current potential* ν_i of a node to an arbitrary value, and assign ν_j of its neighbors according to $\nu_j = \nu_i + y_{ij}$. Repeating this assignment process to next nearest neighbors and so on, the current potentials of all nodes in the tree can be determined. Hence the current potentials can be considered as valid independent variables as the current variables used originally. This implies that we can consider the optimization problem in the space of the current potentials. Since the energy function is invariant under the addition of an arbitrary global constant to the current potentials of all nodes, we introduce an extra regularization term $\epsilon \Sigma_i \mu_i^2 / 2$ to break the translational symmetry, where $\epsilon \to 0$. [Note that the current potentials ν are different from the chemical potentials μ, which are the Lagrange multipliers of the dual formulation in Eq. (2). Only for the quadratic cost $\phi(y) = y^2/2$ can the current be expressed in terms of the difference in chemical potentials. Even in this case, the two potentials may differ by a nonvanishing constant since the resource constraints in Eq. (1) imply that the maximum of the Lagrange multipliers is 0, whereas the current potentials minimize $\epsilon \Sigma_i \nu_i^2 / 2$ and are unlikely to have a maximum value of 0.] The corresponding partition function is

$$Z = \prod_i \int d\nu_i \prod_i \Theta \left[\sum_j \mathcal{A}_{ij}(\nu_j - \nu_i) + \Lambda_i \right]$$

$$\times \exp \left[-\beta \sum_{(ij)} \mathcal{A}_{ij} \phi(\nu_j - \nu_i) - \frac{\beta \epsilon}{2} \sum_i \nu_i^2 \right]. \tag{4}$$

The replicated partition function [1], at a temperature $T \equiv \beta^{-1}$, averaged over all network configurations of connec-

tivity c and capacity distributions $\rho(\Lambda_i)$, is given by

$$\langle Z^n \rangle_{\mathcal{A},\Lambda} = \frac{1}{\mathcal{N}} \sum_{\mathcal{A}_{ij}=0,1} \prod_i \left\{ \delta\left(\sum_j \mathcal{A}_{ij} - c\right) \int d\Lambda_i \rho(\Lambda_i) \right.$$
$$\times \prod_{\alpha=1}^n \left[\int d\nu_i^\alpha \Theta\left(\sum_j \mathcal{A}_{ij}(\nu_j^\alpha - \nu_i^\alpha) + \Lambda_i\right) \right] \Bigg\}$$
$$\times \exp\left(-\beta \sum_{(ij)\alpha} \mathcal{A}_{ij}\phi(\nu_j^\alpha - \nu_i^\alpha) - \frac{\beta\epsilon}{2}\sum_{i\alpha}(\nu_i^\alpha)^2\right).$$
(5)

Here $\mathcal{N} = \sum_{\mathcal{A}_{ij}=0,1}\prod_i \delta(\sum_j \mathcal{A}_{ij} - c)$ is the total number of graphs with connectivity c. This can be easily shown to be [18] $\mathcal{N} = \exp\{N[-(c/2) + (c/2)\ln(cN) - \ln c!]\}$.

The interaction coupling current potentials of different nodes makes it difficult to decouple them in order to define macroscopic order parameters. Nevertheless, additional expansions detailed in Appendix A also show that it is possible to disentangle neighboring node indices. (This justifies the formulation of the optimization in the space of the current potentials $\{\nu_i\}$ rather than that of the currents $\{y_{ij}\}$.) This leads to the following definition of the order parameters:

$$Q_{\mathbf{r},\mathbf{s}} = \frac{1}{\sqrt{cN}}\sum_i z_i \exp\left(\sum_\alpha i\hat{\lambda}_i^\alpha \nu_i^\alpha\right)\prod_\alpha (-i\hat{\lambda}_i^\alpha)^{r_\alpha}(\nu_i^\alpha)^{s_\alpha}, \quad (6)$$

and its conjugate $\hat{Q}_{\mathbf{r},\mathbf{s}}$. Following the analysis of [18] and averaging over the connectivity tensor $\mathcal{A}$ one finds

$$\langle Z^n \rangle_{\mathcal{A},\Lambda} = \exp N\left\{ \frac{c}{2} - c\sum_{\mathbf{r},\mathbf{s}} \hat{Q}_{\mathbf{r},\mathbf{s}} Q_{\mathbf{r},\mathbf{s}} \right.$$
$$+ \ln \int d\Lambda \rho(\Lambda)\prod_\alpha \left(\int d\nu_\alpha \int_{-\Lambda}^\infty d\lambda_\alpha \int \frac{d\hat{\lambda}_\alpha}{2\pi}\right)$$
$$\left. \times \exp\left[\sum_\alpha\left(i\hat{\lambda}_\alpha(\lambda_\alpha + c\nu_\alpha) - \frac{\beta\epsilon}{2}(\nu_\alpha)^2\right)\right]X^c \right\},$$
(7)

where

$$X = \sum_{\mathbf{r},\mathbf{s}} \hat{Q}_{\mathbf{r},\mathbf{s}}\prod_\alpha (-i\hat{\lambda}_\alpha)^{r_\alpha}(\nu_\alpha)^{s_\alpha}$$
$$+ \frac{1}{2}\sum_{\mathbf{r},\mathbf{s}} Q_{\mathbf{r},\mathbf{s}}\prod_\alpha \frac{(\nu_\alpha)^{r_\alpha}}{r_\alpha! s_\alpha!}\left(-i\hat{\lambda}_\alpha - \frac{d}{dy}\right)^{s_\alpha}e^{-\beta\phi(y)}\Bigg|_{y=\nu_\alpha}.$$
(8)

The somewhat unusual indices of the order parameters $Q_{\mathbf{r},\mathbf{s}}$ and $\hat{Q}_{\mathbf{r},\mathbf{s}}$, the vectors $\mathbf{r}$ and $\mathbf{s}$, represent n-component vectors $(r_1,\ldots,r_n)$ and $(s_1,\ldots,s_n)$, respectively. This is a result of the specific interaction considered which entangles nodes of different indices. The order parameters $Q_{\mathbf{r},\mathbf{s}}$ and $\hat{Q}_{\mathbf{r},\mathbf{s}}$ are given by the extremum condition of Eq. (7), i.e., via a set of saddle point equations with respect to the order parameters. To facilitate the solution, we introduce the generating function of $P_{\mathbf{s}}(\mathbf{z})$ and its inversion,

$$P_{\mathbf{s}}(\mathbf{z}) = \sum_{\mathbf{r}} Q_{\mathbf{r},\mathbf{s}}\prod_\alpha \frac{(z_\alpha)^{r_\alpha}}{r_\alpha!},$$

$$Q_{\mathbf{r},\mathbf{s}} = \prod_\alpha \left(\frac{\partial}{\partial z_\alpha}\right)^{r_\alpha} P_{\mathbf{s}}(\mathbf{z})\Bigg|_{\mathbf{z}=0}.$$
(9)

Eliminating $\hat{Q}_{\mathbf{r},\mathbf{s}}$, and substituting the saddle point equation of $Q_{\mathbf{r},\mathbf{s}}$ into $P_{\mathbf{s}}(\mathbf{z})$ in Eq. (9), one finds the recursion relation

$$P_{\mathbf{s}}(\mathbf{z}) = \frac{1}{\mathsf{D}_P}\int d\Lambda \rho(\Lambda)\prod_\alpha \left[\int d\nu_\alpha \int_{-\Lambda}^\infty d\lambda_\alpha \int \frac{d\hat{\lambda}_\alpha}{2\pi}\right.$$
$$\times \exp\left(i\hat{\lambda}_\alpha(\lambda_\alpha + c\nu_\alpha - z_\alpha) - \frac{\beta\epsilon}{2}(\nu_\alpha)^2\right)\Bigg]$$
$$\times \sum_{\mathbf{s}_k}\prod_{k=1}^{c-1} P_{\mathbf{s}_k}(\nu)\prod_{k\alpha} \frac{1}{s_k^\alpha!}\left(-i\hat{\lambda}_\alpha - \frac{d}{dy}\right)^{s_k^\alpha}e^{-\beta\phi(y)}\Bigg|_{y=\nu_\alpha}$$
$$\times \prod_\alpha (\nu_\alpha)^{s_\alpha},$$
(10)

where D_P is a constant given in Eq. (A9). Note that $P_{\mathbf{s}}(\mathbf{z})$ is expressed in terms of $c-1$ functions $P_{\mathbf{s}_k}(\nu)$ ($k=1,\ldots,c-1$), integrated over ν and summed over $\mathbf{s}_k$. This structure is typical of the Bethe lattice description of networks of connectivity c, explained in Sec. IV, where nodes are divided into generations. Each node provides input to an ancestor node and receives input from $c-1$ descendent nodes. This forms a tree structure, in which the state of a node depends on those of its subsequent generations.

In order to derive a set of recursive equations one should make an assumption about the inherent symmetries of the problem. Here we employ the replica symmetric ansatz. In previous treatment of related problems, the order parameters are represented as an integral over moments of the corresponding probability distribution, incorporating the permutation invariance of the replica indices [7,18]. Generalizing to the case of $P_{\mathbf{s}}(\mathbf{z})$, which is an order parameter depending on the continuous variables $\mathbf{z}$, the ansatz takes the form

$$P_{\mathbf{s}}(\mathbf{z}) = \left\langle \prod_\alpha \left(\int d\nu R(z_\alpha, \nu | \mathbf{T})\nu^{s_\alpha}\right)\right\rangle_\Lambda, \quad (11)$$

where $\mathbf{T}$ represents the tree terminated at the vertex node with current potential ν, providing input to the ancestor with chemical potential z, and $\langle \cdots \rangle_\Lambda$ represents the average of the capacities of each node of the tree over the distribution $\rho(\Lambda)$. Note that the replicas are coupled through their common dependence on the quenched variables Λ. This is in contrast to conventional derivations, such as the SK model [1], in which the dependence on the disorder is integrated out, leading to more explicit inter-replica dependencies.

The resultant recursion relation for the function R is independent of the replica indices and hence remains valid in the $n \to 0$ limit. It is given by

$$R(z,\nu|\mathbf{T}) = \frac{1}{\mathsf{D}_R}\prod_{k=1}^{c-1}\left[\int d\nu_k R(\nu,\nu_k|\mathbf{T}_k)\right]$$

$$\times\Theta\left(\sum_{k=1}^{c-1}\nu_k - c\nu + z + \Lambda_{V(\mathbf{T})}\right)$$

$$\times\exp\left[-\frac{\beta\epsilon}{2}\nu^2 - \beta\sum_{k=1}^{c-1}\phi(\nu-\nu_k)\right],\quad (12)$$

where D_R is a constant given in Eq. (A10), and $\mathbf{T}_k$ represents the tree terminated at the kth descendent of the vertex. $\Lambda_{V(T)}$ is the capacity of the vertex of the tree $\mathbf{T}$. Equation (12) expresses $R(z,\nu|\mathbf{T})$ in terms of $c-1$ functions $R(\nu,\nu_k|\mathbf{T}_k)$ $(k=1,\ldots,c-1)$, integrated over ν_k. Again, this is characteristic of the Bethe lattice structure. Furthermore, except for the factor $\exp(-\beta\epsilon\nu^2/2)$, a self-consistent solution of R is that it is a function of $y\equiv\nu-z$, which is interpreted as the current drawn from a node with current potential ν by its ancestor with current potential z. Hence we will express the function R as the product of a *vertex partition function Z_V* and a normalization factor W, which contains any residual dependence on ν. Since ϵ is taken to approach zero in the analysis, it is expected that W should approach a constant independent of ν. Hence we let $R(z,\nu|\mathbf{T})=W(\nu)Z_V(y|\mathbf{T})$. As explained in Appendix A, in the limit $\epsilon\to 0$, the dependence on ν and y decouples; this enables one to derive a recursion relation for the *vertex free energy* [19] $F_V(y|\mathbf{T})\equiv -T\ln Z_V(y|\mathbf{T})$ when a current y is drawn from the vertex of a tree $\mathbf{T}$ [20],

$$F_V(y|\mathbf{T}) = -T\ln\left\{\prod_{k=1}^{c-1}\left(\int dy_k\right)\Theta\left(\sum_{k=1}^{c-1}y_k - y + \Lambda_{V(\mathbf{T})}\right)\right.$$

$$\left.\times\exp\left[-\beta\sum_{k=1}^{c-1}[F_V(y_k|\mathbf{T}_k)+\phi(y_k)]\right]\right\}$$

$$+\left\langle T\ln\left\{\prod_{k=1}^{c}\left(\int dy_k\right)\Theta\left(\sum_{k=1}^{c}y_k+\Lambda_V\right)\right.\right.$$

$$\left.\left.\times\exp\left[-\beta\sum_{k=1}^{c}[F_V(y_k|\mathbf{T}_k)+\phi(y_k)]\right]\right\}\right\rangle_\Lambda.$$

$$(13)$$

In the zero temperature limit, this recursion relation reduces to

$$F_V(y|\mathbf{T}) = \min_{\left\{y_k|\Sigma_{k=1}^{c-1}y_k - y + \Lambda_{V(\mathbf{T})}\geq 0\right\}}\left[\sum_{k=1}^{c-1}[F_V(y_k|\mathbf{T}_k)+\phi(y_k)]\right]$$

$$-\left\langle \min_{\left\{y_k|\Sigma_{k=1}^{c}y_k + \Lambda_V\geq 0\right\}}\left[\sum_{k=1}^{c}[F_V(y_k|\mathbf{T}_k)+\phi(y_k)]\right]\right\rangle_\Lambda.$$

$$(14)$$

The solution of Eqs. (13) and (14) can be obtained numerically in a recursive manner, since the vertex free energy of a

node depends on its own capacity and the disordered configuration of its descendents.

Using the replica approach, and following the derivation of Appendix A, the averaged free energy of the network is given by

$$\langle F\rangle_\Lambda = -N\left\langle T\ln\left\{\prod_{k=1}^{c}\left(\int dy_k\right)\Theta\left(\sum_{k=1}^{c}y_k+\Lambda_V\right)\right.\right.$$

$$\left.\left.\times\exp\left[-\beta\sum_{k=1}^{c}[F_V(y_k|\mathbf{T}_k)+\phi(y_k)]\right]\right\}\right\rangle_\Lambda.\quad (15)$$

The current distribution and the average energy per link can be derived, using the calculated vertex free energy, by integrating the current y' in a link from one vertex to another, fed by the trees $\mathbf{T}_1$ and $\mathbf{T}_2$, respectively; the obtained expressions are $P(y)=\langle\delta(y-y')\rangle_\star$ and $\langle\phi\rangle=\langle\phi(y')\rangle_\star$ where

$$\langle\bullet\rangle_\star = \left\langle\frac{\int dy'\exp[-\beta F_L(y'|\mathbf{T}_1,\mathbf{T}_2)](\bullet)}{\int dy'\exp[-\beta F_L(y'|\mathbf{T}_1,\mathbf{T}_2)]}\right\rangle_\Lambda,\quad (16)$$

with

$$F_L(y'|\mathbf{T}_1,\mathbf{T}_2) = F_V(y'|\mathbf{T}_1) + F_V(-y'|\mathbf{T}_2) + \phi(y').\quad (17)$$

IV. RECURSION RELATION AND FREE ENERGY IN THE BETHE APPROACH

The results in Sec. III can be interpreted using the Bethe approach. Since the connectivity c is low, the probability of finding a loop of finite length on the graph is low, and the Bethe approximation well describes the local environment of a node. In this approximation, a node is connected to c branches in a tree structure, and the correlations among the branches of the tree are neglected. In each branch, nodes are arranged in generations. A node is connected to an ancestor node of the previous generation, and another $c-1$ descendent node of the next generation.

Consider a vertex $V(\mathbf{T})$ of capacity $\Lambda_{V(\mathbf{T})}$, and a current y is drawn from the vertex. At a temperature $T\equiv\beta^{-1}$, one can write an expression for the free energy $F(y|\mathbf{T})$ as a function of the free energies $F(y_k|\mathbf{T}_k)$ of its descendants that branch out from this vertex

$$F(y|\mathbf{T}) = -T\ln\left\{\prod_{k=1}^{c-1}\left(\int dy_k\right)\Theta\left(\sum_{k=1}^{c-1}y_k - y + \Lambda_V\right)\right.$$

$$\left.\times\exp\left[-\beta\sum_{k=1}^{c-1}[F_V(y_k|\mathbf{T}_k)+\phi(y_k)]\right]\right\},\quad (18)$$

where $\mathbf{T}_k$ represents the tree terminated at the kth descendent of the vertex. The free energy can be considered as the sum of two parts

$$F(y|\mathbf{T}) = N_{\mathbf{T}}F_{av} + F_V(y|\mathbf{T}),$$

where $N_{\mathbf{T}}$ is the number of nodes in the tree $\mathbf{T}$, F_{av} is the average free energy per node, and $F_V(y|\mathbf{T})$ is the vertex free energy.

This allows one to decompose the free energy into

$$
\left(\sum_{k=1}^{c-1} N_{\mathbf{T}_k} + 1\right) F_{av} + F_V(y|\mathbf{T})
$$

$$
= \sum_{k=1}^{c-1} N_{\mathbf{T}_k} F_{av} - T \ln\left\{\prod_{k=1}^{c-1}\left(\int dy_k\right)\right.
$$

$$
\times \Theta\left(\sum_{k=1}^{c-1} y_k - y + \Lambda_{V(\mathbf{T})}\right)
$$

$$
\left. \times \exp\left[-\beta\sum_{k=1}^{c-1}[F_V(y_k|\mathbf{T}_k) + \phi(y_k)]\right]\right\}. \quad (19)
$$

To determine the F_{av}, we consider the effects of adding a vertex V which is fed by c individual trees $\mathbf{T}_1,\ldots,\mathbf{T}_c$. The total free energy is then

$$
\left(\sum_{k=1}^{c} N_{\mathbf{T}_k} + 1\right) F_{av}
$$

$$
= -\left\langle T \ln\left\{\prod_{k=1}^{c}\left(\int dy_k\right)\Theta\left(\sum_{k=1}^{c} y_k + \Lambda_V\right)\right.\right.
$$

$$
\left.\left. \times \exp\left[-\beta\sum_{k=1}^{c}(N_{\mathbf{T}_k} F_{av} + F_V(y_k|\mathbf{T}_k) + \phi(y_k))\right]\right\}\right\rangle_\Lambda.
$$
$$(20)$$

Rearranging the terms one obtains a recursion relation identical to Eq. (13). The average free energy per node is given by the second term of Eq. (13) and has the same expression for the free energy as in the replica approach (15).

The recursion relation can also be recast into a form reminiscent of those commonly appearing in Bethe lattices of Ising spin variables, such as in Refs. [6,7,18]. This is achieved by considering the probability distribution of vertex free energies $P[F_V]$. Using Eq. (14),

$$
P[F_V] = \int d\Lambda_V \rho(\Lambda_V)\sum_{k=1}^{c-1}\int \mathcal{D}F_{Vk} P[F_{Vk}]
$$

$$
\times \prod_y \delta\left(-T\ln\left\{\prod_{k=1}^{c-1}\left(\int dy_k\right)\Theta\left(\sum_{k=1}^{c-1} y_k - y + \Lambda_V\right)\right.\right.
$$

$$
\left. \times \exp\left[-\beta\sum_{k=1}^{c-1}[F_{Vk}(y_k) + \phi(y_k)]\right]\right\}
$$

$$
\left. - \langle F\rangle_\Lambda - F_V(y)\right). \quad (21)
$$

Comparing with Bethe lattices of Ising variables, the vertex free energy F_V plays the role of a cavity field. The difference here is that the distribution to be iterated is no longer a function of a single cavity variable. Rather, the distribution is defined in the space of cavity free energy *functions*, since we are dealing with continuous variables. This parallelism enables us to solve the recursion relation by *population dynamics*. At each step of this approach, a new ancestor node is

generated at random, and its vertex free energy is updated.

It is interesting to point out that the iterative equations (13) can be directly linked to those obtained from a principled Bayesian approximation, where the logarithms of the messages passed between nodes are proportional to the vertex free energies. This is shown explicitly in Appendix B.

V. MESSAGE-PASSING ALGORITHM

The local nature of the recursion relation (13) points to the possibility that the network optimization can be solved by message-passing approaches. However, in contrast to other message-passing algorithms which pass conditional probability estimates of *discrete values* to neighboring nodes, the messages in the present context are more complex, since they are *functions* $F_V(y|\mathbf{T})$ of the current y.

The derivation of the algorithm can be viewed as a minimization of the cost function with respect to current changes under the capacity constraint at the neighboring nodes. When the cost is quadratic, the impact of current changes can be described through the first and second derivatives with respect to the vertex free energy. As will be explained at the end of this section, this two-component message is sufficient to provide the exact solution, as long as the algorithm converges.

We follow this route and simplify the message to two parameters, namely, the first and second derivatives of the vertex free energies. Let

$$
(A_{ij}, B_{ij}) \equiv [\partial F_V(y_{ij}|\mathbf{T}_j)/\partial y_{ij}, \partial^2 F_V(y_{ij}|\mathbf{T}_j)/\partial y_{ij}^2]
$$

be the message passed from node j to i. Based on the messages received from the descendents $k \neq i$, the vertex free energy from j to i can be obtained by minimizing the free energy in the space of the current adjustments ε_{jk} drawn from the descendents. Using Eq. (14), we minimize

$$
F_{ij} = \sum_{k\neq i} \mathcal{A}_{jk}\left[A_{jk}\varepsilon_{jk} + \frac{1}{2}B_{jk}\varepsilon_{jk}^2 + \phi'_{jk}\varepsilon_{jk} + \frac{1}{2}\phi''_{jk}\varepsilon_{jk}^2\right], \quad (22)
$$

subject to the constraint

$$
\sum_{k\neq i} \mathcal{A}_{jk}(y_{jk} + \varepsilon_{jk}) - y_{ij} + \Lambda_j \geq 0, \quad (23)
$$

where ϕ'_{jk} and ϕ''_{jk} represent the first and second derivatives of $\phi(y)$ at $y = y_{jk}$, respectively. Introducing the Lagrange multiplier μ_{ij}, the optimal solution is given by

$$
F_{ij}^* = \frac{1}{2}\sum_{k\neq i}\frac{\mathcal{A}_{jk}[\mu_{ij}^2 - (A_{jk} + \phi'_{jk})^2]}{B_{jk} + \phi''_{jk}} \quad (24)
$$

where

$$\mu_{ij} = \min\left[\frac{\sum_{k \neq i} \mathcal{A}_{jk}[y_{jk} - (A_{jk} + \phi'_{jk})(B_{jk} + \phi''_{jk})^{-1}] + \Lambda_j - y_{ij}}{\sum_{k \neq i} \mathcal{A}_{jk}(B_{jk} + \phi''_{jk})^{-1}}, 0\right]. \tag{25}$$

The first and second derivatives of F^*_{ij} with respect to y_{ij} lead to the forward message (A_{ij}, B_{ij}) from node j to i,

$$A_{ij} \leftarrow -\mu_{ij}, \quad B_{ij} \leftarrow \frac{\Theta(-\mu_{ij} + \epsilon)}{\sum_{k \neq i} \mathcal{A}_{jk}(B_{jk} + \phi''_{jk})^{-1}}. \tag{26}$$

We note in passing that when the descendent currents y_{jk} change continuously, both the vertex free energy (24) and the chemical potential (25) change continuously for functions $\phi(y)$ with continuous first derivatives. Hence for the quadratic load balancing task, defined by $\phi(y) = y^2/2$, a self-consistent solution of the recursion relation Eq. (14) consists of vertex free energies which are piecewise quadratic with continuous slopes. This makes the two-parameter message a very precise approximation.

In principle, if the forward messages consist of the full vertex free energy functions, then they are already sufficient for the optimization task. However, since the messages are simplified to be the first and second derivatives of the vertex free energies, each node needs to estimate the optimal currents by approximating the vertex free energy function as a quadratic function. Hence the remaining step of the algorithm is the determination of the drawn current y_{ij} at which the derivatives comprising the messages should be computed. This determination of the *working point* is achieved by passing additional information-provision messages among the nodes, a step not present in conventional message-passing algorithms. The following two methods are proposed for this purpose.

In the first method, when messages are sent from node j to ancestor node i, backward messages y_{jk} computed from the same optimization steps are sent from node j to the descendent nodes $k \neq i$, informing them of the particular arguments to be used for calculating subsequent messages. From Eqs. (22) and (23), this backward message is given by

$$y_{jk} \leftarrow y_{jk} - \frac{A_{jk} + \phi'_{jk} + \mu_{ij}}{B_{jk} + \phi''_{jk}}. \tag{27}$$

In the second method, node j first receives the messages (A_{ji}, B_{ji}) and current y_{ji} from the ancestor node i, and updates the current from y_{ij} to $y_{ij} + \varepsilon_{ij}$ by minimizing the total cost

$$E_{ij} = A_{ij}\varepsilon_{ij} + \frac{1}{2}B_{ij}\varepsilon_{ij}^2 + A_{ji}(-y_{ij} - \varepsilon_{ij} - y_{ji})$$

$$+ \frac{1}{2}B_{ji}(-y_{ij} - \varepsilon_{ij} - y_{ji})^2 + \phi'_{ij}\varepsilon_{ij} + \frac{1}{2}\phi''_{ij}\varepsilon_{ij}^2. \tag{28}$$

In Eq. (28), the first two terms represent the message from i to j, the next two terms from j to i, and the last two terms the

transportation cost in link (ij). The optimal solution is

$$y_{jk} \leftarrow \frac{B_{ij}y_{ij} - A_{ij} - B_{ji}y_{ji} + A_{ji} - \phi'_{ij} + \phi''_{ij}y_{ij}}{B_{ij} + B_{ji} + \phi''_{ij}}. \tag{29}$$

Both methods work well for the quadratic cost function.

Implicit in the information-provision messages is the independent update of the currents y_{ij} and y_{ji} in the opposite directions of the same link. This allows us to use the criterion $y_{ij} = -y_{ji}$ as a check for the convergence of the algorithm. We have used this in our simulation program by requiring the root mean square average of $y_{ij} + y_{ji}$ to be less than a threshold. Another usage of the information-provision messages is in monitoring the optimal cost function during simulations. This saves the extra step of calculating the current associated with a link in the conventional Bethe approach.

An alternative distributed algorithm can be obtained by iterating the chemical potentials of the nodes. The Kühn-Tucker condition requires that the terms $\mu_i(\sum_j \mathcal{A}_{ij}y_{ij} + \Lambda_i)$ in Eq. (2) vanish. Eliminating y_{ij} in terms of the chemical potentials, μ_i can be expressed in terms of μ_j of its neighbors, namely,

$$\mu_i = \min[g_i^{-1}(0), 0]; \quad g_i(x) = \sum_j \mathcal{A}_{ij}[\phi']^{-1}(\mu_j - x) + \Lambda_i. \tag{30}$$

For the quadratic load balancing task, $\phi(y) = y^2/2$ and

$$\mu_i = \min\left[\frac{1}{c}\left(\sum_j \mathcal{A}_{ij}\mu_j + \Lambda_i\right), 0\right]. \tag{31}$$

This provides a local iteration method for the optimization problem. We may interpret this algorithm as a price iteration scheme by noting that the Lagrangian in Eq. (2) can be written as

$$L = \sum_{(ij)} \mathcal{A}_{ij}L_{ij} + \text{const}, \tag{32}$$

where

$$L_{ij} = \phi(y_{ij}) + (\mu_i - \mu_j)y_{ij}. \tag{33}$$

Therefore the problem can be decomposed into independent optimization problems, each for a current on a link. μ_i is the storage price at node i, and each subproblem involves balancing the transportation cost on the link, and the storage cost at node i less that at node j, yielding the optimal solution. This provides a pricing scheme for the individual links to optimize, which simultaneously optimizes the global performance [22].

It can be easily verified that the message-passing algorithm, in the two-parameter approximation, yield solutions

identical to the price iteration algorithm, which is exact, as long as the connectivity is sparse and the algorithms converge. Indeed, simulations provided in Sec. VII show that the two algorithms yield excellent agreement with each other.

One can proceed with the verification by noting from Eq. (27) that $\mu_{ij} = -\phi'_{jk} - A_{jk}$ for all $k \neq i$ at the steady state. Since μ_{ij} is independent of the node i receiving the message, one can write μ_{ij} as μ_j. Similarly, using Eq. (26), $A_{jk} = -\mu_{jk} = -\mu_k$. We then have $\phi'_{jk} = \mu_k - \mu_j$, whose inverse relation is Eq. (3). The resource constraint Eq. (1) then leads to Eq. (30).

The result that the first order message converges to the exact result of the chemical potential at the steady state justifies the simplification of the message to two parameters. It illustrates that higher order messages are not required for the precision of the optimal solution, as long as the algorithm converges. This is natural for the quadratic cost, for which it can be verified that the vertex free energies are piecewise quadratic functions of the currents with continuous slopes. In addition, exact solutions can be found for other cost functions, as described in Sec. VIII. Though the second order messages do not play a role in the final solution, they are useful in tuning the intermediate steps for faster convergence. The situation is reminiscent of the use of both gradient and curvature information in many gradient-based optimization techniques.

VI. HIGH CONNECTIVITY LIMIT

Both the recursive equation (14) and message-passing equations (25) and (26) can be solved numerically as will be shown in the next section. However, scaling laws of the quantities of interest can also be derived analytically in the limit of high connectivity.

We restrict the analysis to the case of quadratic cost function $\phi(y) = y^2/2$. In the limit of large c, Eq. (26) converges to the result

$$B_{ij} = \frac{\Theta(-\mu_{ij})}{c}, \qquad (34)$$

and the currents scale as c^{-1}. Therefore the task of satisfying the capacity constraints is shared by a fraction of $O(1)$ of the descendents. As a result, the collective effects of the descendents on a node can be expressed in terms of the statistical properties of the descendents. Using this scaling property of the currents, Eq. (26) reduces to

$$A_{ij} = \max\left(\frac{1}{c}\left[\sum_{k \neq i} \mathcal{A}_{jk} A_{jk} - \Lambda_j\right], 0\right). \qquad (35)$$

By virtue of the law of large numbers, it is sufficient to consider the mean m_A and variance σ_A^2 of the messages A_{ij}. Respectively, they scale as c^{-1} and c^{-2} with $\Sigma_{k \neq i}\mathcal{A}_{jk}A_{jk}$ being self-averaging. Hence we can write

$$A_{ij} = \frac{1}{c}(cm_A - \Lambda_j)\Theta(cm_A - \Lambda_j). \qquad (36)$$

Averaging over Λ, drawn from a Gaussian of mean $\langle\Lambda\rangle$ and variance 1 used in our numerical studies, one obtains a

self-consistent expression for the parameter $\xi \equiv cm_A - \langle\Lambda\rangle$,

$$\langle\Lambda\rangle = \int_{-\infty}^{\xi} Dz(\xi - z) - \xi = \frac{e^{-\xi^2/2}}{\sqrt{2\pi}} - \xi H(\xi). \qquad (37)$$

A. Current distribution

To obtain the current distribution, one considers the vertex free energies of both ends of a link. For a current y_{ij} flowing from j to i, the total energy is given by

$$E = A_{ij}y_{ij} + \frac{\Theta(-\mu_{ij})}{2c}y_{ij}^2 - A_{ji}y_{ij} + \frac{\Theta(-\mu_{ji})}{2c}y_{ij}^2 + \frac{1}{2}y_{ij}^2, \qquad (38)$$

where we have approximated the working points of the messages to be $y_{ij} = 0$. This is justified since the magnitudes of the messages are $O(c^{-1})$ and $y_{ij} \sim c^{-1}$. Minimizing the energy with respect to the current y_{ij}, one finds

$$y_{ij} = \frac{1}{c}[(cm_A - \Lambda_i)\Theta(cm_A - \Lambda_i) - (cm_A - \Lambda_j)\Theta(cm_A - \Lambda_j)]. \qquad (39)$$

Hence the current distribution is given by

$$P(y) = \int d\Lambda_1 \rho(\Lambda_1) \int d\Lambda_2 \rho(\Lambda_2)$$
$$\times \delta\left\{\frac{1}{c}|(cm_A - \Lambda_1)\Theta(cm_A - \Lambda_1)\right.$$
$$\left. - (cm_A - \Lambda_2)\Theta(cm_A - \Lambda_2)| - y\right\}. \qquad (40)$$

For the Gaussian distribution of capacities, one obtains

$$P(y) = 2\frac{\exp\left(-\frac{c^2y^2}{4}\right)}{\sqrt{4\pi/c^2}}H\left(\frac{cy - 2\xi}{\sqrt{2}}\right) + 2H(\xi)\frac{\exp\left(-\frac{(cy - \xi)^2}{2}\right)}{\sqrt{2\pi/c^2}}$$
$$+ H(\xi)^2\delta(y). \qquad (41)$$

This shows that the distribution $P(cy)/c$, obtained by rescaling the argument by c^{-1}, is independent of c, and depends solely on the average capacity $\langle\Lambda\rangle$ through ξ. In particular, the fraction of *idle* links is given by

$$P(y = 0) = H(\xi)^2. \qquad (42)$$

The physical picture of this scaling behavior is that the total current required by a node to satisfy its capacity constraint is shared by the links.

B. Average energy

Using Eq. (39), the average energy per link can be written as

$$\langle\phi\rangle = \frac{1}{c^2}\{\langle(cm_A - \Lambda)^2\Theta(cm_A - \Lambda)\rangle$$

$$- \langle(cm_A - \Lambda)\Theta(cm_A - \Lambda)\rangle^2\}. \tag{43}$$

For the Gaussian capacity distribution, it becomes

$$\langle\phi\rangle = \frac{1}{c^2}[I_2(\xi) - I_1(\xi)^2], \tag{44}$$

where

$$I_1(\xi) = \int_{-\infty}^{\xi} Dz(\xi - z) = \frac{e^{-\xi^2/2}}{\sqrt{2\pi}} + \xi H(-\xi), \tag{45}$$

$$I_2(\xi) = \int_{-\infty}^{\xi} Dz(\xi - z)^2 = \xi\frac{e^{-\xi^2/2}}{\sqrt{2\pi}} + (\xi^2 + 1)H(-\xi). \tag{46}$$

We are also interested in how the energy is distributed in the network. Consider the average energy per link $\langle\phi|\Lambda\rangle$ among those links connected to nodes of capacity Λ. Using Eq. (39),

$$\langle\phi|\Lambda\rangle = \frac{1}{c^2}\int d\Lambda_2\rho(\Lambda_2)[(cm_A - \Lambda)\Theta(cm_A - \Lambda)$$

$$\times(cm_A - \Lambda_2)\Theta(cm_A - \Lambda_2)]^2. \tag{47}$$

For the Gaussian capacity distribution, this becomes

$$\langle\phi|\Lambda\rangle = \frac{1}{2c^2}\{I_2(\xi) - [I_1(\xi)^2 - \Lambda^2]\Theta[I_1(\xi) - \Lambda]\}. \tag{48}$$

C. Chemical potential distribution

To obtain the distribution of the chemical potentials, one follows a similar treatment and considers a central node 0 fed by c descendents. Introducing a Lagrange multiplier to enforce the capacity constraint, one replaces the energy minimization problem by the Lagrangian

$$L = \sum_{j=1}^{c}\left[A_{0j}y_{0j} + \frac{\Theta(-\mu_{0j})}{2c}y_{0j}^2 + \frac{1}{2}y_{0j}^2\right] + \mu\left(\sum_{j=1}^{c}y_{0j} + \Lambda_0\right). \tag{49}$$

The currents are given by $y_{0j} = -A_{0j} - \mu$, and the chemical potential by

$$\mu = \min\left(\frac{1}{c}\left[\Lambda_0 - \sum_{j=1}^{c}A_{0j}\right], 0\right).$$

In the large c limit, the approximated expression for μ becomes

$$\mu = \frac{1}{c}(\Lambda_0 - cm_A)\Theta(cm_A - \Lambda_0), \tag{50}$$

and the chemical potential distribution is similarly derived,

$$P(\mu) = \int_{-\infty}^{cm_A} d\Lambda\rho(\Lambda)\delta\left[\frac{1}{c}(\Lambda - cm_A) - \mu\right]$$

$$+ \int_{cm_A}^{\infty} d\Lambda\rho(\Lambda)\delta(\mu). \tag{51}$$

For the Gaussian capacity distribution, it reduces to

$$P(\mu) = \frac{\exp\left(-\frac{(c\mu + \xi)^2}{2}\right)}{\sqrt{2\pi/c^2}}\Theta(-\mu) + H(\xi)\delta(\mu). \tag{52}$$

This shows that the distribution $P(c\mu)/c$, obtained by rescaling the argument by c^{-1}, is independent of c and depends solely on the average capacity $\langle\Lambda\rangle$ through ξ. In particular, the fraction of unsaturated nodes is given by

$$P(\mu = 0) = H(\xi). \tag{53}$$

D. Resource distribution

We define the resource at a node i by

$$r_i \equiv \Lambda_i + \sum_j \mathcal{A}_{ij}y_{ij}. \tag{54}$$

The currents are obtained by minimizing

$$E_i = \sum_j \mathcal{A}_{ij}\left[A_{ij}y_{ij} + \frac{\Theta(-\mu_j)}{2c}y_{ij}^2 + \frac{1}{2}y_{ij}^2\right] \tag{55}$$

subject to the constraints $\sum_j\mathcal{A}_{ij}y_{ij} + \Lambda_i \geq 0$. Introducing the Lagrange multiplier μ_i for the constraint, we obtain

$$r_i = \max[\Lambda_i - cm_A, 0]. \tag{56}$$

Hence the resource distribution is given by

$$P(r) = \int_{cm_A}^{\infty} d\Lambda\rho(\Lambda)\delta(\Lambda - cm_A - r) + \int_{-\infty}^{cm_A} d\Lambda\rho(\Lambda)\delta(r). \tag{57}$$

For the Gaussian capacity distribution, it reduces to

$$P(r) = \frac{\exp\left[-\frac{1}{2}(r - \xi)^2\right]}{\sqrt{2\pi}}\Theta(r) + H(-\xi)\delta(r). \tag{58}$$

This shows that the resource distribution becomes independent of c in the large c limit, confirming the picture that the current in a link scales as c^{-1}, summing up to a total current of c^0 satisfying the resource requirement of the nodes.

E. Dynamics

To analyze the dynamics in the limit of large c, one considers random sequential updates using the algorithm presented in Sec. V. Time is divided into steps of size $\Delta t = 1/cN$. At each time step, a directed link from node j to i is randomly chosen, such that each directed link is chosen exactly once in each integer interval of time, and the messages of the links are updated.

The current y_{jk}, for a link feeding node j, is updated in the backward messages corresponding to the forward ones from j to $i \neq k$. (This implies that y_{jk} is updated K times in a time step. As will be shown, the algorithm uses information updated in the previous step to compute the optimal current. Since the previous step lies in the previous interval, this approach is not the most efficient for monitoring the evolving average energy.)

Denote the average of message $A_{jk}(t)$ over all links at time t as $m_A(t)$, and $\hat{m}_A(t)$ the expected value of $A_{jk}(t)$ when it is updated at time t. Then, for a time t in the interval between t_0 and t_0+1 this leads to the dynamical equation

$$\frac{dm_A(t)}{dt} = \hat{m}_A(t) - m_A(t_0) \quad \text{for } t_0 \leqslant t < t_0 + 1. \quad (59)$$

Suppose the link ij is updated at time t, according to Eq. (26). Then the average over link ij becomes

$$c\hat{m}_A(t) = \int_{-\infty}^{cm_A(t)} d\Lambda \rho(\Lambda)[cm_A(t) - \Lambda]. \quad (60)$$

For the Gaussian capacity distribution, this becomes

$$c\hat{m}_A(t) = \int_{-\infty}^{\xi(t)} Dz(\xi(t) - z) = I_1(\xi(t)), \quad (61)$$

where $\xi(t) = cm_A(t) - \langle \Lambda \rangle$. It is convenient to convert Eq. (59) to a dynamical equation for $\xi(t)$,

$$\frac{d\xi(t)}{dt} = I_1(\xi(t)) - \xi(t_0) - \langle \Lambda \rangle \quad \text{for } t_0 \leqslant t < t_0 + 1,$$

with the initial condition $\xi(0) = -\langle \Lambda \rangle$.

The dynamics of the average energy depends on whether one adopts the backward or forward information-provision method, described by Eqs. (27) and (29), respectively. We first consider the case of backward information provision. Suppose the link from j to i is updated at time t_{ij} in the interval between t_0 and t_0+1. Using Eq. (27),

$$y_{jk}(t_{ij}) = -A_{jk}(t_{jk}^-) - \mu_{ij}(t_{ij}), \quad (62)$$

where $A_{jk}(t_{jk}^-)$ is given by Eq. (36) and t_{jk}^- is the instant that the link from k to j is previously chosen for update. With probability $t_0 + 1 - t_{ij}$, t_{jk}^- lies in the previous time interval between $t_0 - 1$ and t_0. Otherwise, t_{jk}^- lies between t_0 and t_{ij}.

To calculate the average energy $\langle \phi(t_0+1) \rangle$, one can express $\langle y_{jk}^2 \rangle$ in terms of the moments $\langle A_{jk}^2(t_{jk}^-) \rangle$, $\langle \mu_{ij}^2(t_{ij}) \rangle$, and $\langle A_{jk}(t_{jk}^-)\mu_{ij}(t_{ij}) \rangle$. Hence on averaging over all descendents k, denoted as $\langle \ \rangle_k$, the second moment of $A_{jk}(t_{jk}^-)$ is given via Eq. (36) by

$$\langle A_{jk}(t_{jk}^-)^2 \rangle_k = \frac{1}{c^2} \left[\int_{t_0-1}^{t_0} dt_{jk}^- (t_0 + 1 - t_{ij}) + \int_{t_0}^{t_{ij}} dt_{jk}^- \right]$$
$$\times \int d\Lambda \rho(\Lambda)[cm_A(t_{jk}^-) - \Lambda]^2 \Theta[cm_A(t_{jk}^-) - \Lambda]. \quad (63)$$

For the Gaussian capacity distribution, this becomes

$$\langle A_{jk}(t_{jk}^-)^2 \rangle_k = \frac{1}{c^2} \bar{I}_2(t_{ij}), \quad (64)$$

where

$$\bar{I}_l(t) \equiv \left[\int_{t_0-1}^{t_0} dt'(t_0 + 1 - t) + \int_{t_0}^{t} dt' \right] I_l(\xi(t')). \quad (65)$$

Averaging over (ij) at time t_0+1,

$$\langle A_{jk}(t_{jk}^-)^2 \rangle_{ijk} = \frac{1}{c^2} \int_{t_0}^{t_0+1} dt \bar{I}_2(t), \quad (66)$$

which can be simplified to

$$\langle A_{jk}(t_{jk}^-)^2 \rangle_{ijk} = \frac{1}{c^2} \left[\frac{1}{2} \int_{t_0-1}^{t_0} dt I_2(\xi(t)) + \int_{t_0}^{t_0+1} dt(t_0 + 1 - t)I_2(\xi(t)) \right]. \quad (67)$$

A similar calculation follows for the second moment of $\mu_{ij}(t_{ij})$ at time t_0+1, leading to

$$\langle \mu_{ij}(t_{ij})^2 \rangle_{ij} = \frac{1}{c^2} \int_{t_0}^{t_0+1} dt I_2(\xi(t)) \quad (68)$$

for the Gaussian capacity distribution.

Similarly, the expression for the crossed moment in the case of Gaussian capacity distribution is

$$\langle A_{jk}(t_{jk}^-)\mu_{ij}(t_{ij}) \rangle_{ijk} = -\frac{1}{c^2} \int_{t_0}^{t_0+1} dt \bar{I}_1(t)I_1(\xi(t)), \quad (69)$$

which can be simplified to

$$\langle A_{jk}(t_{jk}^-)\mu_{ij}(t_{ij}) \rangle_{ijk} = -\frac{1}{c^2} \left\{ \left[\int_{t_0-1}^{t_0} dt I_1(\xi(t)) \right] \right.$$
$$\times \left[\int_{t_0}^{t_0+1} dt(t_0 + 1 - t)I_1(\xi(t)) \right]$$
$$\left. + \frac{1}{2} \left[\int_{t_0}^{t_0+1} dt I_1(\xi(t)) \right]^2 \right\}. \quad (70)$$

Hence the average energy per link in the case of Gaussian capacity distribution is

$$\langle \phi(t_0 + 1) \rangle = \frac{1}{2c^2} \left\{ \frac{1}{2} \int_{t_0-1}^{t_0} dt I_2(\xi(t)) \right.$$
$$+ \int_{t_0}^{t_0+1} dt(t_0 + 1 - t)I_2(\xi(t))$$
$$- 2 \left[\int_{t_0-1}^{t_0} dt I_1(\xi(t)) \right]$$
$$\times \left[\int_{t_0}^{t_0+1} dt(t_0 + 1 - t)I_1(\xi(t)) \right]$$

$$-\left[\int_{t_0}^{t_0+1} dtI_1(\xi(t))\right]^2 + \int_{t_0}^{t_0+1} dtI_2(\xi(t))\Bigg\}.$$

$$(71)$$

Next, we consider the case of forward information-provision described by Eq. (29), whose large c limit is given by

$$y_{ij}(t_{ij}) = -A_{ij}(t_{ij}^-) + A_{ji}(t_{ji}^-), \qquad (72)$$

where the link from j to i is updated at time t_{ij} in the time interval between t_0 and t_0+1, t_{ij}^- and t_{ji}^- are, respectively, the instants that the links from j to i and from i to j are previously chosen for update. Derivation analogous to the backward information-provision method yields

$$\langle\phi(t_0+1)\rangle = \frac{1}{2c^2}\Bigg\{\int_{t_0-1}^{t_0} dtI_2(\xi(t)) - 2\left[\int_{t_0-1}^{t_0} dtI_1(\xi(t))\right]$$
$$\times\left[\int_{t_0}^{t_0+1} dt\bar{I}_1(t)\right] + \int_{t_0}^{t_0+1} dt\bar{I}_2(t)\Bigg\}, \qquad (73)$$

which can be simplified to

$$\langle\phi(t_0+1)\rangle = \frac{1}{2c^2}\Bigg\{\frac{3}{2}\int_{t_0-1}^{t_0} dtI_2(\xi(t))$$
$$+ \int_{t_0}^{t_0+1} dt(t_0+1-t)I_2(\xi(t))$$
$$- 2\left[\int_{t_0-1}^{t_0} dtI_1(\xi(t))\right]$$
$$\times\left[\int_{t_0}^{t_0+1} dt(t_0+1-t)I_1(t)\right]$$
$$- \left[\int_{t_0-1}^{t_0} dtI_2(\xi(t))\right]^2\Bigg\}. \qquad (74)$$

Equation (72) shows that the forward information-provision method uses only outdated information to calculate the current. Consequently, the convergence of the average energy is slower than that of the backward information-provision method by about half a step.

VII. NUMERICAL RESULTS

To examine the accuracy of the theoretical results and the efficacy of the message-passing algorithm of Sec. V we conducted a series of numerical experiments. First, we solved numerically the recursive equation (14) using population dynamics for various connectivity values and obtained various quantities of interest from it, including the energy, current, and chemical potential distribution. Second, we carried out simulations using the algorithms of Eqs. (26) and (25) and compared the results to those obtained from the numerical simulations. We then compared the scaling properties of the results with respect to the connectivity with the theoretical scaling obtained in Sec. VI. All experiments in this section have been carried out using the quadratic cost $\phi(y)=y^2/2$.

To solve numerically the recursive equation (14) we have discretized the vertex free energy functions $F_V(y|\mathbf{T})$ into a vector, whose ith component is the value of the function corresponding to the current y_i. To speed up the optimization search at each node, we first find the *vertex saturation current* drawn from a node such that: (a) the current drawn by each descendent node separately optimizes its own vertex free energy plus the transportation cost to the node being considered. For descendent nodes k, this current y_k^* is given by

$$y_k^* = \mathrm{argmin}_{y_k}[F_V(y_k|\mathbf{T}_k) + \phi(y_k)]. \qquad (75)$$

(b) The resource of the node considered is just used up. For node j, its vertex saturation current y_j^s is given by

$$y_j^s = \sum_{k\neq j} \mathcal{A}_{jk}y_k^* + \Lambda_j. \qquad (76)$$

For current below this saturation point, the vertex free energy remains constant. That is, $F_V(y_j|\mathbf{T}_j)=F_V(y_j^s|\mathbf{T}_j)$ for $y_j \leq y_j^s$. Hence this provides a convenient starting point for searching the optimal solutions. The drawn current can then be increased in small discrete steps, and the optimal solution can be found, for example, using an exhaustive search, by varying the descendent currents in small discrete steps. This approach is particularly convenient for $c=3$, where the search is confined to a single parameter. For larger values of c, other search techniques, such as conjugate gradient search, can be used.

These recursive equations provide a discretized representation of the vertex free energy $F_V(y|\mathbf{T})$, from which various properties of the system can be calculated.

Average energy. To compute the average energy, we randomly draw $c-1$ nodes, compute the optimal current flowing between them, and repeat the sampling to obtain the average.

The results of iteration for a Gaussian distribution $\rho(\Lambda)$ with variance 1 and average $\langle\Lambda\rangle$ were described in [12]. There we found that the convergence rate slows down when $\langle\Lambda\rangle$ decreases towards 0. A cusp in the relaxation rate dependence on $\langle\Lambda\rangle$ exists at $\langle\Lambda\rangle\approx 0.45$, where the fraction of saturated nodes is about 0.48, close to the percolation threshold of 0.5 for $c=3$. Hence the cusp is probably related to the appearance of a percolating cluster of negative resources, which draws currents from increasingly extensive regions of nodes with excess resources to satisfy the demand.

Dependence on the connectivity. We have presented in [12] evidence that the currents scale as c^{-1}, so that after rescaling, the average energy $c^2\langle\phi\rangle$ [see Fig. 1 (inset)], the current distribution $P(cy)/c$, and the resource distribution $P(r)$ become principally dependent on the average capacity $\langle\Lambda\rangle$, and only weakly dependent on the connectivity c. The scaling property extends to the dynamics of the optimization process. All results of increasing connectivity approach those of the high connectivity limit derived in Sec. VI.

Here we further study how the high connectivity limit is approached from increasing finite values of c. We define the empirical scaling factor s by
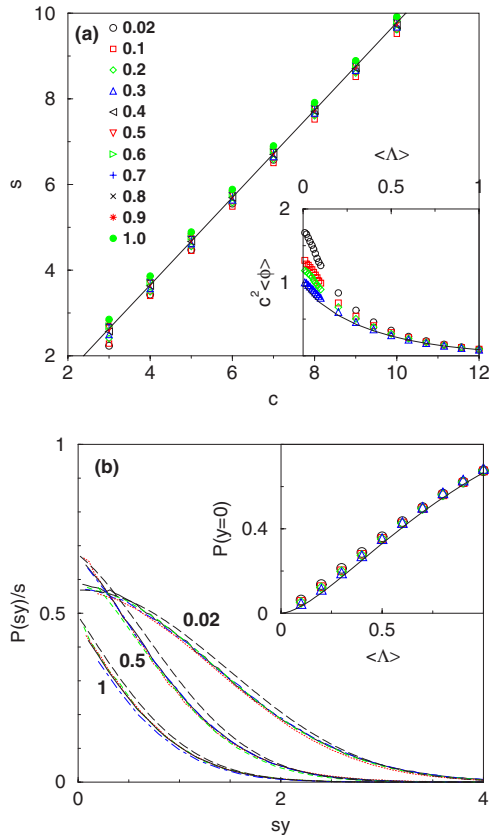
FIG. 1. (Color online) Results for $N=1000$ and $\phi(y)=y^2/2$. (a) The dependence of the empirical scaling factor s on the connectivity c for different values of $\langle\Lambda\rangle$. Line: best fit. Inset: $c^2\langle\phi\rangle$ as a function of $\langle\Lambda\rangle$. Symbols: $c=3$ ($\bigcirc$), 4 ($\square$), 5 ($\diamond$), 10 ($\triangle$), and high c (line). (b) The continuous component of the current distribution $P(sy)/s$ for $\langle\Lambda\rangle=0.02$, 0.5, and 1. Lines: $c=3$ (solid), 4 (dotted), 5 (dashed), 10 (dot-dashed), and high c (long dashed). Inset: $P(y=0)$ as a function of $\langle\Lambda\rangle$, symbols: same as (a) inset.

$$s = \sqrt{\frac{\lim_{c\to\infty} c^2\langle\phi\rangle}{\langle\phi\rangle}}. \qquad (77)$$

s is expected to converge to c in the high connectivity limit. As shown in Fig. 1, the empirical scaling factor corresponding to different values of $\langle\Lambda\rangle$ collapse on a linear curve with slope 1. The best fit is $s\approx 1.02c-0.43$. It is remarkable that the network behavior already converges to scaling at low values of c.

We make use of this empirical scaling factor to rescale the current distribution. The current distribution consists of a delta function component at $y=0$ [Fig. 1(b) (inset) [21]] and a continuous component, whose breadth decreases with $\langle\Lambda\rangle$. Excluding the delta function component, the continuous distribution after rescaling is shown in Fig. 1(b). The approach to the high connectivity limit is even faster when compared with that by setting the scaling factor to be $c-1$ [12].

Energy dependence on node capacity. We divide the nodes into ten groups according to their node capacities. Nodes in group 1 have their capacities among the top 10%, those in group 2 the next highest 10%, and so on. For each group, we then calculate the average energy per link for
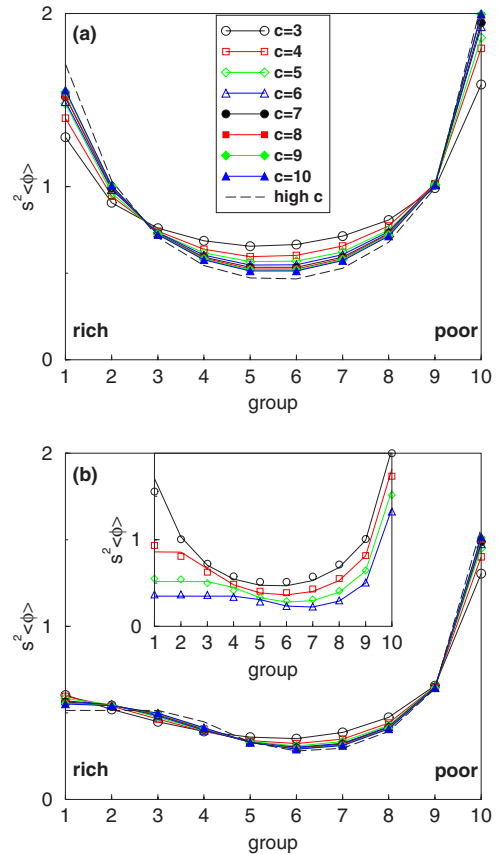

FIG. 2. (Color online) Results for $N=1000$ and $\phi(y)=y^2/2$. (a) The rescaled energy per link connected to the ten groups of nodes with decreasing ranges of capacities at $\langle\Lambda\rangle=0.02$ for different connectivities indicated in the legend. (b) Same as (a) but at $\langle\Lambda\rangle=0.2$. Inset: Comparison of the curves for $\langle\Lambda\rangle=0.02$, 0.1, 0.2, and 0.3 (top to bottom). Lines: high c. Symbols: $c=10$ and $\langle\Lambda\rangle=0.02$ ($\bigcirc$), 0.1 ($\square$), 0.2 ($\diamond$), and 0.3 ($\triangle$).

those links connected to the nodes of that group. The general trend can be seen in Figs. 2(a) and 2(b). Group 1 consists of the richest nodes. Since they are the resource providers to the rest of the network, their connected links have high average energy. On the other hand, group 10 consists of the poorest nodes. Since they are the resource consumers of the network, their connected links also have high average energy. Compared with group 1, their average energy is even higher due to the enforcement of the resource constraints, Eq. (1). By comparison, the groups in between consist of relay nodes which typically receive resources from the richer ones and provide resources to the poorer ones. The energies of their connected links have intermediate averages. Figures 2(a) and 2(b) show that these different roles played by nodes of different capacities can lead to a significant difference in the average energies of their connected links.

Furthermore, when one compares networks of different connectivities, one finds that the rescaled energy curves become only weakly dependent on the connectivity. The convergence to the high connectivity limit is rather fast.

Figure 2(b) (inset) shows the rescaled energy curves in the high connectivity limit for different $\langle\Lambda\rangle$. When $\langle\Lambda\rangle$ increases, a plateau starts to develop among the groups of
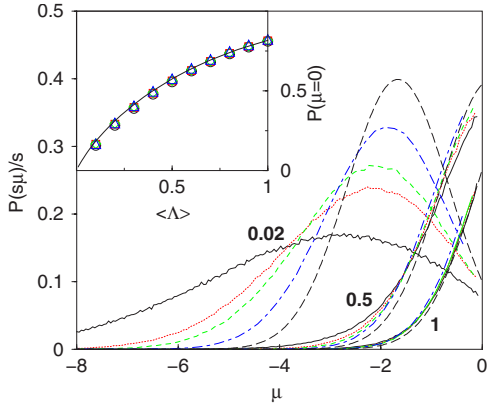
FIG. 3. (Color online) Results for system size $N=1000$ and $\phi(y)=y^2/2$. The chemical potential distribution $P(s\mu)/s$ for $\langle\Lambda\rangle=0.02$, 0.5, and 1. Lines: $c=3$ (solid), 4 (dotted), 5 (dashed), 10 (dot-dashed), and high c (long dashed). Inset: $P(\mu=0)$ as a function of $\langle\Lambda\rangle$. Symbols: $c=3$ ($\bigcirc$), 4 ($\square$), 5 ($\diamond$), 10 ($\triangle$), and high c (long dashed).

richer nodes, indicating that the rich nodes become unsaturated in their resource provision, so that the energy of their connected links becomes independent of their excess resources. They have $\Lambda>I_1(\xi)$ according to Eq. (48). Simulation results for $c=10$ presented in the same figure provide support to this behavior. In fact, the development of this plateau is already visible in the simulation results of finite connectivities in Fig. 3, whose trend shows that the homogenization of energy among the links connected to the rich nodes is increasingly effective when the connectivity increases.

Chemical potential distribution. Both the message-passing and price iteration algorithms allow us to study the distribution $P(\mu)$ of the chemical potentials μ. $P(\mu)$ consists of a delta function at $\mu=0$ [Fig. 3 (inset)] and a continuous component. The width of the continuous component increases when $\langle\Lambda\rangle$ decreases. Note the concurrence of low average resources and highly negative values of μ, consistent with the economic interpretation of μ as the storage cost of a node. The scaling property of the distribution is illustrated in Fig. 3. For $\langle\Lambda\rangle=0.5$ and 1, the distributions collapse well even for relatively low values of c. For $\langle\Lambda\rangle=0.02$, a considerable dependence on c remains after rescaling. However, the approach to the high c limit is visible.

VIII. GENERAL COST FUNCTIONS

The cost used so far was the, rather simple, quadratic cost. In this section we examine the applicability of the message-passing algorithm for more general costs. Two representative costs will be studied.

(a) The *anharmonic* cost function is used to model the effects of costs rising faster than quadratic,

$$\phi(y) = \frac{y^2}{2} + \frac{u|y|^3}{3}. \tag{78}$$

(b) The *frictional* cost function is used to model the ef-

fects of frictional forces in resource allocation, which add an extra cost per unit current in a link irrespective of direction. Hence it is also useful in networks with multiple classes of traffic sharing the same links. The cost function takes the form

$$\phi(y) = \frac{y^2}{2} + v|y|. \tag{79}$$

Note that these cost functions represent two distinct cases. The former has well-defined first and second derivatives for all of its arguments. In the latter case, the frictional cost function does not have a second derivative at $y=0$. There is a kink in the cost function at the point of zero current, thus increasing the preference for idle links, or equivalently the reluctance to switch on a link. As will be shown, the convergence of the message-passing algorithm is much more difficult and modifications are necessary.

A. Anharmonic cost

1. Price iteration

Introducing Lagrange multipliers for the capacity constraints, the function to be minimized is

$$L = \sum_{(ij)} \left(\frac{y_{ij}^2}{2} + \frac{u|y_{ij}|^3}{3} \right) + \sum_i \mu_i \left(\sum_{j\in\mathcal{L}_i} y_{ij} + \Lambda_i \right), \tag{80}$$

where $\mathcal{L}_i$ is the set of neighboring nodes of i. Optimizing with respect to $y_{ij}\equiv-y_{ji}$, one obtains the relation

$$y_{ij} = \frac{1}{u}\left[\sqrt{\frac{1}{4} + u|\mu_j - \mu_i|} - \frac{1}{2} \right]\mathrm{sgn}(\mu_j - \mu_i). \tag{81}$$

Using the capacity constraints, the chemical potential μ_i is given by $\mu_i=\min[g_i^{-1}(0),0]$, where g_i^{-1} is the inverse of the function

$$g_i(x) = \sum_{j\in\mathcal{L}_i} \frac{1}{u}\left[\sqrt{\frac{1}{4} + u|\mu_j - x|} - \frac{1}{2} \right]\mathrm{sgn}(\mu_j - x) + \Lambda_i. \tag{82}$$

This provides a price iteration scheme. We solve this equation using the bisection method, noting that the function is monotonic nonincreasing. This requires one to know the solution bounds. Let $\mu_{\max}$ and $\mu_{\min}$ be the maximum and minimum of the chemical potentials among the neighbors of node i. Examining the cases of $\Lambda_i>0$ and $\Lambda_i<0$ separately, one finds the range for the solution of x,

$$\mu_{\min} + \min\left[\frac{\Lambda_i}{c}\left(1 + \frac{u|\Lambda_i|}{c}\right), 0 \right]$$
$$\leq x \leq \min\left\{ \mu_{\max} + \max\left[\frac{\Lambda_i}{c}\left(1 + \frac{u|\Lambda_i|}{c}\right), 0 \right], 0 \right\}. \tag{83}$$

2. Message passing

Since the cost function has well-defined first and second derivatives for all of its arguments, the message-passing al-
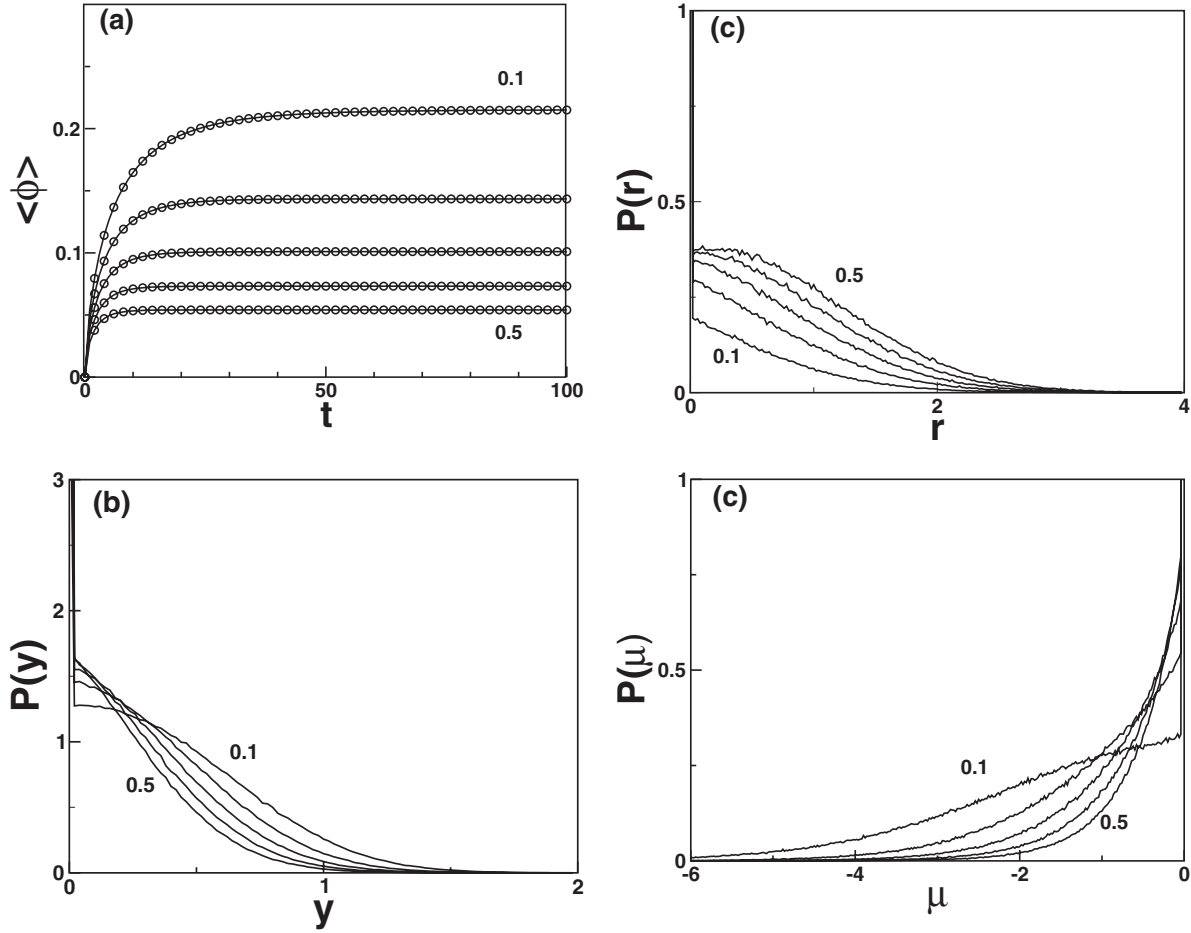
FIG. 4. Results for $N=1000$, $c=3$, the anharmonic cost function with $u=1$, and 1000 samples. (a) $\langle\phi\rangle$ obtained by the price iteration algorithm, utilizing Eqs. (82) and (83), as a function of t for $\langle\Lambda\rangle=0.1$, 0.2, 0.3, 0.4, and 0.5 (top to bottom). (b) The corresponding current distribution $P(y)$. (c) The corresponding resource distribution $P(r)$. (d) The corresponding chemical potential distribution $P(\mu)$.

gorithm formulated in Sec. V is directly applicable:

$$A_{ij} \leftarrow -\mu_{ij}, \qquad (84)$$

$$B_{ij} \leftarrow \frac{\Theta(-\mu_{ij})}{\sum\limits_{k\in\mathcal{L}_j\backslash\{i\}}(B_{jk}+\phi''_{jk})^{-1}}, \qquad (85)$$

where

$$\mu_{ij} = \min\left\{\left[\sum_{k\in\mathcal{L}_j\backslash\{i\}}[y_{jk}-(A_{jk}+\phi'_{jk})(B_{jk}+\phi''_{jk})^{-1}]+\Lambda_j-y_{ij}\right]\right.$$
$$\left.\times\left[\sum_{k\in\mathcal{L}_j\backslash\{i\}}(B_{jk}+\phi''_{jk})^{-1}\right]^{-1},0\right\}, \qquad (86)$$

and the backward message is given by

$$y_{jk} \leftarrow y_{jk} - \frac{A_{jk}+\phi'_{jk}+\mu_{ij}}{B_{jk}+\phi''_{jk}}. \qquad (87)$$

For the anharmonic cost function,

$$\phi'_{jk} = y_{jk} + u y^2_{jk}\,\mathrm{sgn}\,y_{jk} \quad \text{and} \quad \phi''_{jk} = 1 + 2u|y_{jk}|.$$

3. Simulation results

To study the behavior of the various algorithms in the case of the anharmonic cost function, in comparison to the quadratic cost, we carried out a simulation under similar conditions to those of Sec. VII.

Figure 4(a) shows the average energy per link as a function of iteration steps of the price iteration algorithm for the anharmonic cost function. Figures 4(b)–4(d) show the distributions, $P(y)$, $P(r)$, and $P(\mu)$ of the currents, resources, and chemical potentials, respectively, at the corresponding values of $\langle\Lambda\rangle$. The results obtained are very similar to those of the quadratic cost function and show the same qualitative behavior, as can be seen by comparing Figs. 4(a)–4(d) with Figs. 1(a) (inset), 2(a), 2(b), and 3(a), respectively.

In Figs. 5(a)–5(d) we compare the behavior of the price iteration and message-passing algorithms by plotting the average energy per link, the fraction of idle links $P(y=0)$, the fraction of saturated nodes $P(r=0)$, and the fraction of unsaturated nodes $P(\mu=0)$, respectively, as a function of $\langle\Lambda\rangle$ for both algorithms at the anharmonic strengths $u=1$ and 3. Both methods converge to the same value throughout the range examined and for the two u values examined.

To provide a more quantitative comparison of the cost functions, we also plotted in Figs. 5(b)–5(d) $P(y=0)$, $P(r$
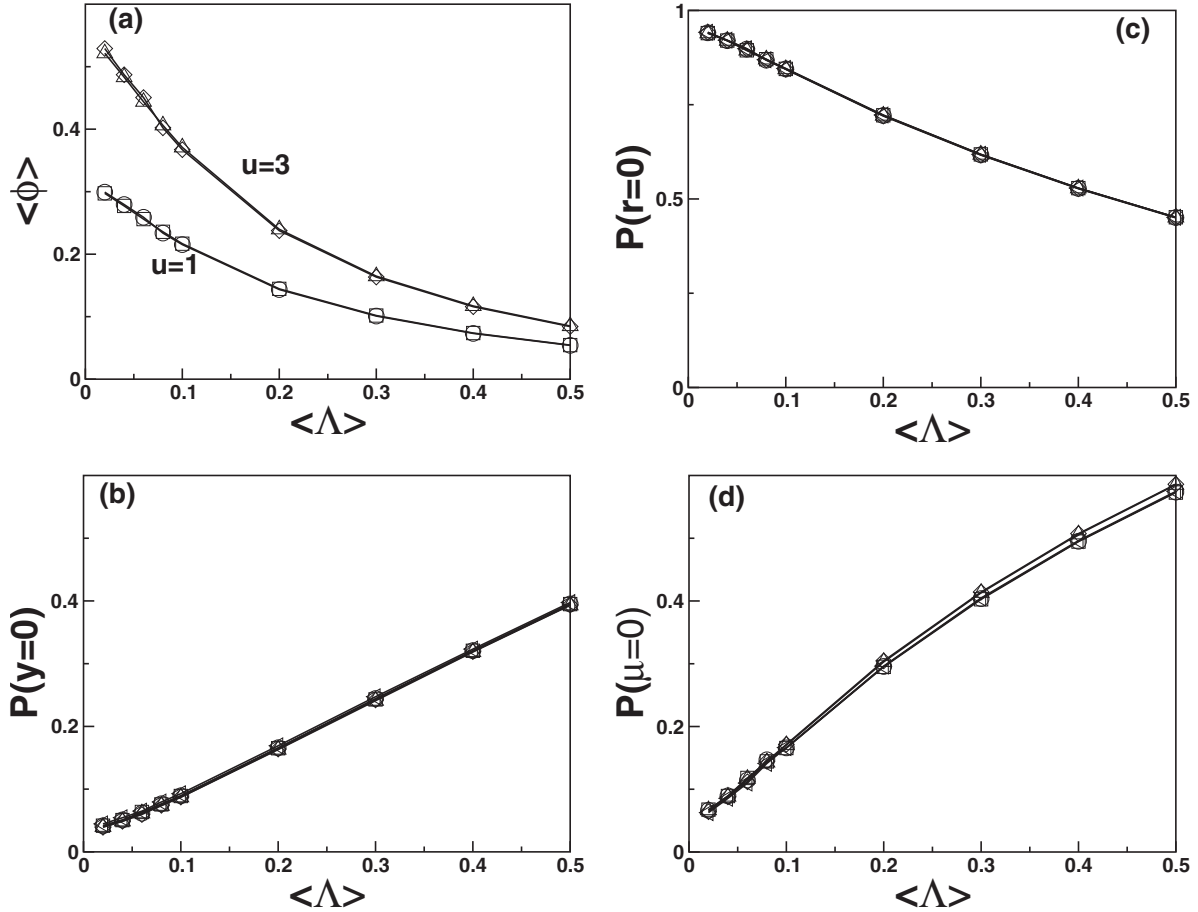
FIG. 5. Results for $N=1000$, $c=3$, the anharmonic cost function, and 1000 samples. (a) Average energy per link $\langle \phi \rangle$. (b) The fraction of idle links $P(y=0)$. (c) The fraction of saturated nodes $P(r=0)$. (d) The fraction of unsaturated nodes $P(\mu=0)$. Symbols in (a)–(d): results obtained by the price iteration algorithm for the quadratic cost function ($u=0$) ($\triangleleft$), and the anharmonic cost function with $u=1$ ($\bigcirc$) and $u=3$ ($\diamond$); results obtained by the message-passing algorithm are also shown for $u=1$ ($\square$) and $u=3$ ($\triangle$).

$=0$), and $P(\mu=0)$, respectively, for the quadratic ($u=0$) cost function. Simulations have been carried out under the same conditions ($N=1000$, $c=3$, and 1000 samples). It is remarkable that there is little difference between the quadratic and anharmonic cases. The different cost functions merely change the continuous components of these distributions, leaving the delta function components effectively unchanged.

B. Frictional cost

1. Price iteration

Introducing Lagrange multipliers for the capacity constraints, the function to be minimized is

$$L = \sum_{(ij)} \left(\frac{y_{ij}^2}{2} + v|y_{ij}| \right) + \sum_i \mu_i \left(\sum_{j \in \mathcal{L}_i} y_{ij} + \Lambda_i \right). \quad (88)$$

Optimizing with respect to $y_{ij} \equiv -y_{ji}$, one obtains

$$y_{ij} = [\mu_j - \mu_i - v \, \text{sgn}(\mu_j - \mu_i)]\Theta[|\mu_j - \mu_i| - v]. \quad (89)$$

Using the capacity constraints, the chemical potential is

given by $\mu_i = \min[g_i^{-1}(0), 0]$, where g_i^{-1} is the inverse of the function

$$g_i(x) = \sum_{j \in \mathcal{L}_i} [\mu_j - x - v \, \text{sgn}(\mu_j - x)]\Theta[|\mu_j - x| - v] + \Lambda_i. \quad (90)$$

Since $g_i(x)$ is monotonic nonincreasing and piecewise linear, we have a fast way to solve the equation by finding the function at its $2c$ turning points, located at $x = \mu_j \pm v$. If $g_i(\mu_{\min} - v) < 0$, the solution is given by $(\mu_{\min} - v) - g_i(\mu_{\min} - v)/c$; if $g_i(\mu_{\min} - v) \geq 0$, then among the turning points with $g_i(x) \geq 0$, one finds the one with the minimum value of $g_i(x)$, and the solution is given by $x - g_i(x)/g_i'(x^+)$.

2. Message passing

Message-passing algorithms for the currents have not been successful in this case, presumably due to the divergence of the second derivative at $y=0$. This in turn requires some form of regularization that causes the effects of friction to be exhibited in the first, but not the second, derivative in finite systems. This inconsistency prevents the algorithm from converging.

We present here an approach based on the chemical potential representation. To formulate an appropriate version of message passing for this case, we return to the minimization of the energy of Sec. V, namely,

$$E_j^{\backslash i} = \sum_{k \in \mathcal{L}_j \backslash \{i\}} \left[A_{jk} \varepsilon_{jk} + \frac{1}{2} B_{jk} \varepsilon_{jk}^2 + \frac{1}{2} (y_{jk} + \varepsilon_{jk})^2 + v|y_{jk} + \varepsilon_{jk}| \right], \tag{91}$$

subject to the constraints

$$\sum_{k \in \mathcal{L}_j \backslash \{i\}} (y_{jk} + \varepsilon_{jk}) - y_{ij} + \Lambda_j \geqslant 0, \tag{92}$$

introduced by employing the Lagrange multipliers μ_{ij}. The optimal solution is given by $\mu_{ij} = \min[g_{ij}^{-1}(0), 0]$, where g_{ij}^{-1} is the inverse of the function

$$g_{ij}(x) = \sum_{k \in \mathcal{L}_j \backslash \{i\}} (1 + B_{jk})^{-1}[B_{jk} y_{jk} - A_{jk} - x$$
$$- v \, \mathrm{sgn}(B_{jk} y_{jk} - A_{jk} - x)]$$
$$\times \Theta[|B_{jk} y_{jk} - A_{jk} - x| - v] + \Lambda_i - y_{ij}. \tag{93}$$

The forward messages become

$$A_{ij} \leftarrow -\mu_{ij},$$

$$B_{ij} \leftarrow \frac{\Theta(-\mu_{ij})}{\displaystyle\sum_{k \in \mathcal{L}_j \backslash \{i\}} (1 + B_{jk})^{-1} \Theta[|B_{jk} y_{jk} - A_{jk} - \mu_{ij}| - v]}. \tag{94}$$

To complete the algorithm, we need information-provision messages to determine the drawn current y_{ij} at which the messages should be computed. Analogous to the case of quadratic cost functions, two methods are proposed.

In the method of backward information-provision messages, the backward messages are computed directly from the optimization of Eq. (91) and sent from node j to the descendent nodes, namely,

$$y_{jk} \leftarrow (1 + B_{jk})^{-1}[B_{jk} y_{jk} - A_{jk} - \mu_{ij}$$
$$- v \, \mathrm{sgn}(B_{jk} y_{jk} - A_{jk} - \mu_{ij})]$$
$$\times \Theta[|B_{jk} y_{jk} - A_{jk} - \mu_{ij}| - v]. \tag{95}$$

This algorithm reduces the error at steady state to a level that is still rather high. A careful examination of the solution shows that the error is contributed by oscillatory solutions between y_{ij} and y_{ji}. Hence a learning rate η is introduced:

$$y_{jk} \leftarrow (1 - \eta) y_{jk} + \eta (1 + B_{jk})^{-1}[B_{jk} y_{jk} - A_{jk} - \mu_{ij}$$
$$- v \, \mathrm{sgn}(B_{jk} y_{jk} - A_{jk} - \mu_{ij})]$$
$$\times \Theta[|B_{jk} y_{jk} - A_{jk} - \mu_{ij}| - v]. \tag{96}$$

The case $\eta = 1$ corresponds to the original algorithm.

In the method of forward information-provision messages, a node first receives the messages from the ancestor immediately before it updates its messages. The working point is obtained by minimizing the energy

$$E_{ij} = A_{ij} \varepsilon_{ij} + \frac{1}{2} B_{ij} \varepsilon_{ij}^2 + A_{ji}(-y_{ij} - \varepsilon_{ij} - y_{ji})$$
$$+ \frac{1}{2} B_{ji}(-y_{ij} - \varepsilon_{ij} - y_{ji})^2 + \frac{1}{2}(y_{ij} + \varepsilon_{ij})^2 + v|y_{ij} + \varepsilon_{ij}|, \tag{97}$$

with the optimal solution

$$y_{ij} \leftarrow (1 + B_{ij} + B_{ji})^{-1}[B_{ij} y_{ij} - A_{ij} - B_{ji} y_{ji} + A_{ji}$$
$$- v \, \mathrm{sgn}(B_{ij} y_{ij} - A_{ij} - B_{ji} y_{ji} + A_{ji})]$$
$$\times \Theta(|B_{ij} y_{ij} - A_{ij} - B_{ji} y_{ji} + A_{ji}| - v). \tag{98}$$

For further improvement, a learning rate is introduced, namely,

$$y_{ij} \leftarrow (1 - \eta) y_{ij} + \eta (1 + B_{ij} + B_{ji})^{-1}[B_{ij} y_{ij} - A_{ij} - B_{ji} y_{ji} + A_{ji}$$
$$- v \, \mathrm{sgn}(B_{ij} y_{ij} - A_{ij} - B_{ji} y_{ji} + A_{ji})]$$
$$\times \Theta(|B_{ij} y_{ij} - A_{ij} - B_{ji} y_{ji} + A_{ji}| - v). \tag{99}$$

3. Simulation results

To study the behavior of both price iteration and message-passing algorithms in the case of the frictional cost function, we carried out simulations under similar conditions to those of Sec. VII. Figure 6(a) shows the average energy per link as a function of iteration steps of the price iteration algorithm. Figures 6(b)–6(d) show the current, resource, and chemical potential distributions, $P(y)$, $P(r)$, and $P(\mu)$, respectively, for the various $\langle \Lambda \rangle$ values.

The results shown in Figs. 6(a)–6(c) exhibit a similar qualitative behavior to those of the quadratic and anharmonic cost functions. However, there is a substantial difference in the chemical potential distribution, shown in Fig. 6(d) as a pseudogap develops in the range $v < \mu < 0$, as well as a kink at $\mu = -2v$. From Eq. (89) one notices that a link becomes idle when the potential difference at its vertices is less than v, accounting for the existence of the pseudogap.

A quantitative comparison between the results obtained by price iteration (90) and message-passing [Eqs. (94) and (95)] algorithms ($\eta = 1$, no learning rate) is presented in Fig. 7. A comparison of the average energy per link as a function of $\langle \Lambda \rangle$, the fraction of idle links $P(y=0)$, and the fraction of unsaturated nodes $P(\mu=0)$ is shown in Figs. 7(a)–7(d), respectively, showing good agreement between the result obtained using both algorithms. Results obtained by both price iteration and message-passing algorithms for a friction ($v = 1$) cost are also contrasted with results obtained for the quadratic ($v=0$) cost in Figs. 7(b) and 7(d).

As shown in Fig. 7, the price iteration and the original message-passing algorithms yield results agreeing in the average energy (a), the fraction of idle links (b), and the fraction of unsaturated nodes (d). Compared with the quadratic cost function, the fraction of idle links is considerably increased after introducing the friction, as shown in Fig. 7(b). However, as shown in Fig. 7(c), the message-passing algorithm gives values much lower than those of price iteration, and is inconsistent with the results in Fig. 7(d).
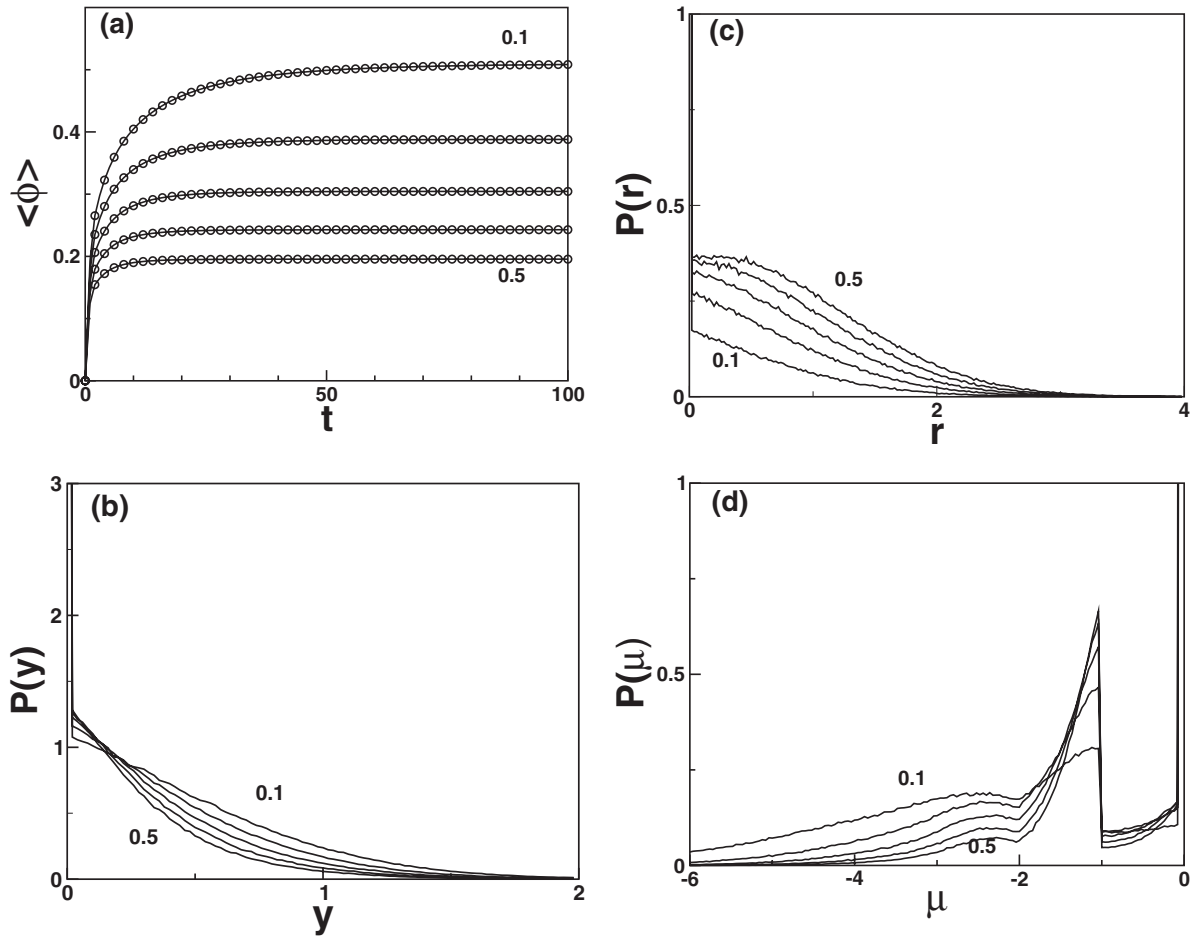
FIG. 6. Results for $N=1000$, $c=3$, the frictional cost function with $v=1$, and 1000 samples. (a) $\langle\phi\rangle$ obtained by the price iteration algorithm as a function of t for $\langle\Lambda\rangle=0.1$, 0.2, 0.3, 0.4, and 0.5 (top to bottom). (b) The corresponding current distribution $P(y)$. (c) The corresponding resource distribution $P(r)$. (d) The corresponding chemical potential distribution $P(\mu)$.

The resource distribution in Fig. 8(a) explains the discrepancy. Compared with the results in Fig. 6(c), the sharp peak at $r=0$ is broadened to finite values of r. This shows that the original message-passing algorithm is not precise in computing the resources. Furthermore, the chemical potential distribution in Fig. 8(b) exhibits rough features in the pseudogap, and the jumps near the edge of the pseudogap are less sharp than those in Fig. 6(d).

These unsatisfactory performances of the message-passing algorithm can be traced to its nonconvergence. In message passing, convergence is monitored by the root-mean-square average of $\langle[(y_{ij}+y_{ji})/2]^2\rangle^{1/2}$, which is expected to approach 0. As shown in Fig. 8(c) for the original algorithm ($\eta=1$), the convergence parameter reaches 0.04 at $t=500$, compared with the value of 0.0003 for the price iteration algorithm.

To improve convergence, we introduce a learning rate according to Eqs. (96) and (99). As shown in Figs. 8(c) and 8(d), convergence improves for decreasing η, but is also slowed down. Comparing the two information-provision methods, the one using forward information-provision messages converges faster.

As shown in Fig. 9(a), better convergence is obtained by the forward information-provision messages in 500 steps.

Figure 9(b) summarizes the improvement in the fraction of saturated nodes on introducing the learning rate for 500 steps; results obtained using the price iteration algorithm are provided for comparison. Obviously, further improvement can be made by increasing the number of time steps, and hence depends on the amount of computational resources one wishes to commit.

IX. CONCLUSION

The paper presents a study of inference and optimization tasks of real value edges on sparse graphs under given constraints and cost measure. A generic framework comprising of sparsely connected nodes, representing constraints, and edges representing current variables connecting them, is used as the basic framework for the inference and optimization tasks. Inference of real values attributed to the graph edges has very rarely been studied before within and outside the statistical physics community. Although both theoretical analysis and algorithmic solutions can be obtained for any connectivity profile, we restricted this study to the case of fixed connectivity, c.

The framework is analyzed using both the replica method and Bethe approximation to obtain a set of recursive equa-
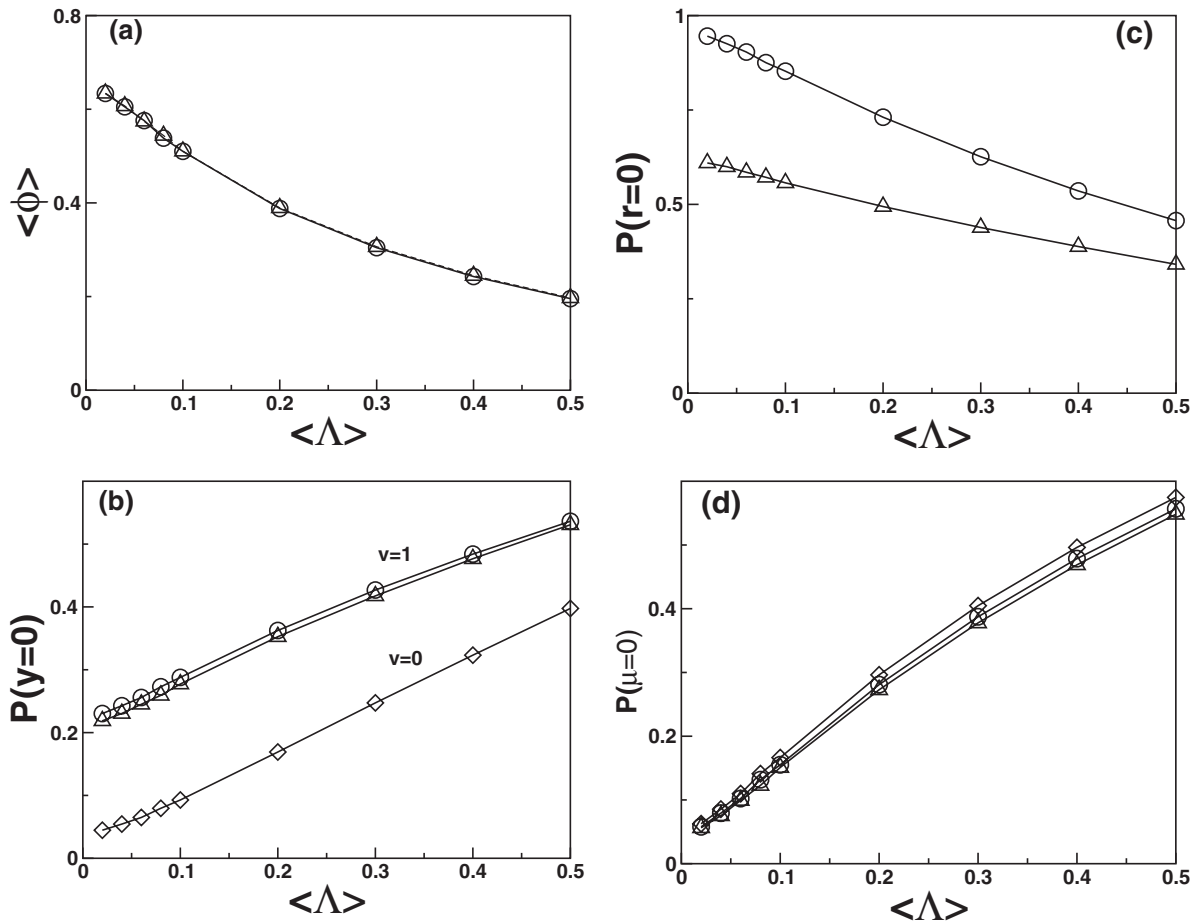
FIG. 7. Results for $N=1000$, $c=3$, the frictional cost function with $v=1$, and 1000 samples. (a) Average energy $\langle \phi \rangle$. (b) The fraction of idle links $P(y=0)$. (c) The fraction of unsaturated nodes $P(\mu=0)$. (d) The fraction of saturated nodes $P(r=0)$. Symbols in (a)–(d): results obtained for the frictional cost function by the price iteration algorithm ($\bigcirc$) and the message-passing algorithm ($\square$); results obtained for the quadratic cost function ($v=0$) ($\diamondsuit$).

tions to be solved numerically. The solutions provide numerical results for the free energy, average energy, and the distribution of currents, resources, and chemical potentials for the various cases. The recursive equations also enabled us to obtain scaling rules for various quantities of interest as a function of the node connectivity. In addition, we have devised message-passing and price iteration algorithms for solving the optimization problem. The message-passing algorithm is based on passing first and second derivatives of the vertex free energy, representing the local contribution to the system's free energy, thus saving the need to pass the full free energy functions as messages. Despite the simplicity of the two-parameter messages, they yield exact solutions in the limit of sparse connectivity as long as they converge.

Most numerical studies have been carried out for the case of quadratic cost that corresponds to the resource allocation problem which initiated this study. In this case we fixed the nodes' capacities, representing biases in the local constraints, to quenched variables drawn from some Gaussian distribution of given mean $\langle \Lambda \rangle$ and unit variance. Numerical results for various parameters values, c and $\langle \Lambda \rangle$, show excellent agreement between the analytical and algorithmic approaches both for finite and asymptotic connectivity values. Moreover, they expose an interesting percolation transition

of the clusters of nodes with negative resources when $\langle \Lambda \rangle$ varies, giving rise to a slowing down of the convergence of the saddle point equations below a certain value of $\langle \Lambda \rangle$.

To study the efficacy of our approach to other cost measures we have examined two different costs that include anharmonic and friction terms. We have applied two different algorithms in these cases based on the price iteration and message passing. Price iterations involve solving a nonlinear equation of the chemical potential at each step numerically. On the other hand, message passing involves updating the messages based on the working points estimated from the information-provision messages. While the obtained solutions are qualitatively similar to that of the quadratic cost, there are also substantial algorithmic and conceptual differences, especially in the case of friction cost. For the optimization task studied in this paper, price iteration is simpler in implementation and converges better when compared with message passing. However, for future extensions to inference problems at finite temperatures, we expect the message-passing approach to be more appropriate. It is also useful to adopt an adaptive learning rate as a function of time to optimize the performance [24].

We believe this research opens a rich area for further investigations with many potential applications, especially
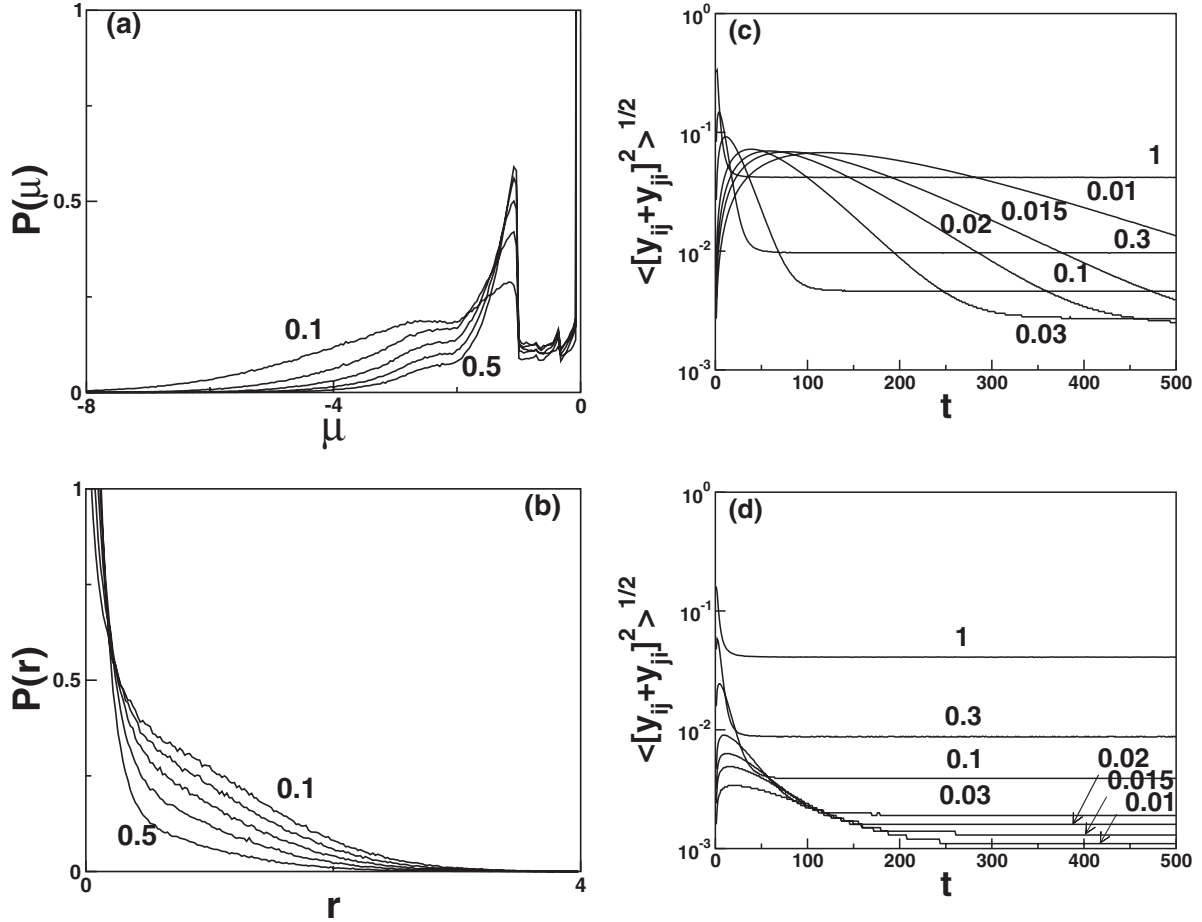
FIG. 8. Results for $N=1000$, $c=3$, the frictional cost function with $v=1$, and 1000 samples. (a) The chemical potential distribution $P(\mu)$ for $\langle\Lambda\rangle=0.1$, 0.2, 0.3, 0.4, and 0.5 (top to 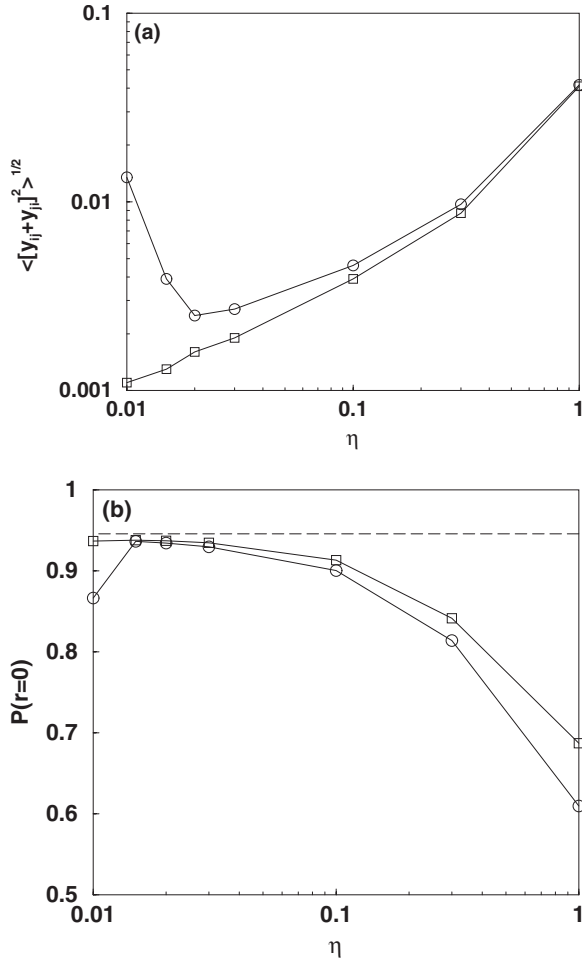bottom). (b) The corresponding resource distribution $P(r)$. (c) and (d) The convergence parameter $\langle[(y_{ij}+y_{ji})/2]^2\rangle^{1/2}$ of the message-passing algorithm as a function of iteration steps and for various η values, using backward and forward information-provision messages in (c) and (d), respectively.

when additional restrictions are imposed and other costs considered. More specifically, one may consider bandwidth limited links [23] and other nonlinear costs which are of interest in realistic networks. We expect that many nonlinear costs may exhibit replica symmetry-breaking effects, and it would be interesting to consider how the analyses and algorithms should be modified to cope with these effects.

ACKNOWLEDGMENTS

APPENDIX A: REPLICA APPROACH TO NETWORK OPTIMIZATION

To calculate the averaged replicated partition function (5), we employ an integral representation of the step function to obtain

$$\langle Z^n\rangle = \frac{1}{\mathcal{N}}\sum_{\mathcal{A}_{ij}=0,1}\prod_i\oint\frac{dz_i}{2\pi iz_i^{c+1}}\prod_i z_i^{\Sigma_j\mathcal{A}_{ij}}$$
$$\times\prod_i\left[\int d\Lambda_i\rho(\Lambda_i)\prod_\alpha\int d\nu_i^\alpha\int_{-\Lambda_i}^\infty d\lambda_i^\alpha\int\frac{d\hat\lambda_i^\alpha}{2\pi}\right.$$
$$\left.\times e^{i\hat\lambda_i^\alpha\lambda_i^\alpha-\beta\epsilon(\nu_i^\alpha)^2/2}\right]\exp\left[-\sum_{i\alpha}i\hat\lambda_i^\alpha\sum_j\mathcal{A}_{ij}(\nu_j^\alpha-\nu_i^\alpha)\right.$$
$$\left.-\beta\sum_{(ij)}\sum_\alpha\mathcal{A}_{ij}\phi(\nu_j^\alpha-\nu_i^\alpha)\right]. \quad (A1)$$

Collecting terms containing $\mathcal{A}_{ij}$ and summing over them, one obtains

$$\langle Z^n\rangle = \frac{1}{\mathcal{N}}\prod_i\oint\frac{dz_i}{2\pi iz_i^{c+1}}\prod_i\left[\int d\Lambda_i\rho(\Lambda_i)\right.$$
$$\left.\times\prod_\alpha\int d\nu_i^\alpha\int_{-\Lambda_i}^\infty d\lambda_i^\alpha\int\frac{d\hat\lambda_i^\alpha}{2\pi}e^{i\hat\lambda_i^\alpha\lambda_i^\alpha-\beta\epsilon(\nu_i^\alpha)^2/2}\right]$$
$$\times\prod_{(ij)}\left[1+z_iz_j\exp\left(\sum_\alpha(i\hat\lambda_i^\alpha-i\hat\lambda_j^\alpha)(\nu_i^\alpha-\nu_j^\alpha)\right.\right.$$

which gives rise to the mean-field parameters

$$Q_{\mathbf{r},\mathbf{s}}^m = \frac{1}{\sqrt{cN}}\sum_i z_i^m \exp\left(\sum_\alpha im\hat{\lambda}_i^\alpha \nu_i^\alpha\right)\prod_\alpha (-im\hat{\lambda}_i^\alpha)^{r_\alpha}(\nu_i^\alpha)^{s_\alpha}$$

(A4)

and the conjugate parameters $\hat{Q}_{\mathbf{r},\mathbf{s}}^m$. The replicated and averaged partition function $\langle Z^n\rangle$ becomes

$$\langle Z^n\rangle = \frac{1}{\mathcal{N}}\prod_{\mathbf{r},\mathbf{s},m}\int \frac{d\hat{Q}_{\mathbf{r},\bar{s}}^m dQ_{\mathbf{r},\bar{s}}^m}{2\pi/cN}\exp\left(-cN\sum_{\mathbf{r},\mathbf{s},m}\hat{Q}_{\mathbf{r},\bar{s}}^m Q_{\mathbf{r},\bar{s}}^m\right)$$

$$\times\prod_i\left[\oint\frac{dz_i}{2\pi i z_i^{c+1}}\int d\Lambda_i \rho(\Lambda_i)\right.$$

$$\times\prod_\alpha\left(\int d\nu_i^\alpha \int_{-\Lambda_i}^\infty d\lambda_i^\alpha \int\frac{d\hat{\lambda}_i^\alpha}{2\pi}e^{i\hat{\lambda}_i^\alpha\lambda_i^\alpha-\beta\epsilon(\nu_i^\alpha)^2/2}\right)\right]$$

$$\times\exp\left\{\sqrt{cN}\sum_{\mathbf{r},\mathbf{s},m}\hat{Q}_{\mathbf{r},\mathbf{s}}^m\sum_i z_i^m \exp\left(\sum_\alpha im\hat{\lambda}_i^\alpha\nu_i^\alpha\right)\right.$$

$$\times\prod_\alpha(-im\hat{\lambda}_i^\alpha)^{r_\alpha}(\nu_i^\alpha)^{s_\alpha}+\sqrt{cN}\sum_{\mathbf{r},\mathbf{s},m}\frac{(-)^{m-1}}{2m}\frac{Q_{\mathbf{r},\mathbf{s}}^m}{\prod_\alpha r_\alpha! s_\alpha!}$$

$$\times\sum_j z_j^m \exp\left(\sum_\alpha im\hat{\lambda}_j^\alpha\nu_j^\alpha\right)\prod_\alpha(\nu_j^\alpha)^{r_\alpha}$$

$$\times\left.\left(-im\hat{\lambda}_j^\alpha-\frac{d}{dy}\right)^{s_\alpha}e^{-m\beta\phi(y)}\bigg|_{y=\nu_j^\alpha}\right\}.$$

(A5)

The integration over z_i is dominated by the term $m=1$ in the cth order expansion of the exponential term that leads to Eq. (7). Both $Q_{\mathbf{r},\mathbf{s}}$ and $\hat{Q}_{\mathbf{r},\mathbf{s}}$ are then given by the saddle point equations

$$Q_{\mathbf{r},\mathbf{s}} = \frac{N_1}{D}\quad\text{and}\quad \hat{Q}_{\mathbf{r},\mathbf{s}} = \frac{N_2}{D}$$

where

$$N_1 = \int d\Lambda\rho(\Lambda)\prod_\alpha \int d\nu_\alpha \int_{-\Lambda}^\infty d\lambda_\alpha \int\frac{d\hat{\lambda}_\alpha}{2\pi}$$

$$\times\exp\left[\sum_\alpha\left(i\hat{\lambda}_\alpha(\lambda_\alpha+c\nu_\alpha)-\frac{\beta\epsilon}{2}(\nu_\alpha)^2\right)\right]X^{c-1}$$

$$\times\prod_\alpha(-i\hat{\lambda}_\alpha)^{r_\alpha}(\nu_\alpha)^{s_\alpha},$$

FIG. 9. The convergence parameter $\langle[(y_{ij}+y_{ji})/2]^2\rangle^{1/2}$ of message-passing algorithms as a function of the learning rate η at $t=500$ with $N=1000$, $c=3$, the frictional cost function with $v=1$, and 1000 samples. Symbols: information provision using backward messages ($\bigcirc$) and forward messages ($\square$). (b) The corresponding fraction of saturated nodes. Symbols: same as in (a), and the price iteration algorithm (dashed line).

$$-\beta\sum_\alpha\phi(\nu_j^\alpha-\nu_i^\alpha)\bigg)\bigg].$$

(A2)

This includes a mixed term of i and j indices. An additional expansion is required to disentangle the two indices. The product over (ij) can be written as an exponential function whose argument is

$$\sum_{ij}\sum_{m=1}^\infty\frac{(-)^{m-1}}{2m}z_i^m z_j^m\exp\left(\sum_\alpha im\hat{\lambda}_i^\alpha\nu_i^\alpha\right)$$

$$\times\exp\left(\sum_\alpha im\hat{\lambda}_j^\alpha\nu_j^\alpha\right)\sum_{r_\alpha,s_\alpha,t_\alpha}\prod_\alpha\frac{(-im\hat{\lambda}_i^\alpha\nu_j^\alpha)^{r_\alpha}}{r_\alpha!}$$

$$\times\frac{(-im\hat{\lambda}_j^\alpha\nu_i^\alpha)^{s_\alpha}(\nu_i^\alpha)^{t_\alpha}}{s_\alpha!}\frac{}{t_\alpha!}\left(-\frac{d}{dy}\right)^{t_\alpha}e^{-m\beta\phi(y)}\bigg|_{y=\nu_j^\alpha},$$

(A3)

$$N_2 = \frac{1}{2} \int d\Lambda \rho(\Lambda) \prod_\alpha \int d\nu_\alpha \int_{-\Lambda}^\infty d\lambda_\alpha \int \frac{d\hat{\lambda}_\alpha}{2\pi}$$

$$\times \exp\left[\sum_\alpha \left(i\hat{\lambda}_\alpha(\lambda_\alpha + c\nu_\alpha) - \frac{\beta\epsilon}{2}(\nu_\alpha)^2\right)\right] X^{c-1}$$

$$\times \prod_\alpha \frac{(\nu_\alpha)^{r_\alpha}}{r_\alpha!} \frac{1}{s_\alpha!} \left(-i\hat{\lambda}_\alpha - \frac{d}{dy}\right)^{s_\alpha} e^{-\beta\phi(y)}\bigg|_{y=\nu_\alpha},$$

$$D = \int d\Lambda \rho(\Lambda) \prod_\alpha \int d\nu_\alpha \int_{-\Lambda}^\infty d\lambda_\alpha \int \frac{d\hat{\lambda}_\alpha}{2\pi}$$

$$\times \exp\left[\sum_\alpha \left(i\hat{\lambda}_\alpha(\lambda_\alpha + c\nu_\alpha) - \frac{\beta\epsilon}{2}(\nu_\alpha)^2\right)\right] X^c, \quad \text{(A6)}$$

where X is given by Eq. (8). By virtue of the saddle point equations (A6), one can show that

$$\hat{Q}_{\mathbf{r,s}} = \frac{1}{2} \sum_{\mathbf{t,u}} \prod_\alpha \frac{(-)^{t_\alpha}}{r_\alpha! t_\alpha! (s_\alpha - t_\alpha)! u_\alpha!}$$

$$\times \left(\frac{d}{dy}\right)^{t_\alpha + u_\alpha} e^{-\beta\phi(y)}\bigg|_{y=0} Q_{\mathbf{s-t,r+u}}. \quad \text{(A7)}$$

Exploiting the even nature of $\phi(y)$ and relation (A7) [20], the expressions for X and $Q_{\mathbf{r,s}}$ reduce to

$$X = \sum_{\mathbf{r,s}} \frac{Q_{\mathbf{r,s}}}{\prod_\alpha r_\alpha! s_\alpha!} \prod_\alpha (\nu_\alpha)^{r_\alpha} \left(-i\hat{\lambda}_\alpha - \frac{d}{dy}\right)^{s_\alpha} e^{-\beta\phi(y)}\bigg|_{y=\nu_\alpha}.$$

$$\text{(A8)}$$

To better understand the symmetry properties of the order parameters, we consider the generating function $P_{\mathbf{s}}(\mathbf{z})$ and its inversion in Eq. (9). Substituting Eq. (A8) into Eq. (9), we reproduce Eq. (10), with D_P being

$$D_P = \int d\Lambda \rho(\Lambda) \prod_\alpha \left[\int d\nu_\alpha \int_{-\Lambda}^\infty d\lambda_\alpha \int \frac{d\hat{\lambda}_\alpha}{2\pi}\right]$$

$$\times \exp\left(i\hat{\lambda}_\alpha(\lambda_\alpha + c\nu_\alpha) - \frac{\beta\epsilon}{2}(\nu_\alpha)^2\right) \sum_{\mathbf{s}_k} \prod_{k=1}^c P_{\mathbf{s}_k}(\nu)$$

$$\times \prod_{k\alpha} \frac{1}{s_k^\alpha!} \left(-i\hat{\lambda}_\alpha - \frac{d}{dy}\right)^{s_k^\alpha} e^{-\beta\phi(y)}\bigg|_{y=\nu_\alpha}. \quad \text{(A9)}$$

Once we have represented the order parameters $Q_{\mathbf{r,s}}$ using the generating function $P_{\mathbf{s}}(\mathbf{z})$, we can make explicit assumptions about their symmetry properties. In particular, in the replica symmetric ansatz, we consider functions of the form Eq. (11).

Notice that the replicas in Eq. (11) are coupled through their common dependence on the disordered distribution of Λ. This is different from the SK model, in which the dependence on the disorder is integrated out, and the interaction between the replicas is explicit. Using the ansatz (11), the recursion relation for $P_{\mathbf{s}}(\mathbf{z})$ can be replaced by a recursion relation for the function R in Eq. (12), where

$$D_R = \left\langle \left\{ \int d\nu \prod_{k=1}^c \left[\int d\nu_k R(\nu, \nu_k | \mathbf{T}_k)\right]\right.\right.$$

$$\times \Theta\left(\sum_{k=1}^c \nu_k - c\nu + \Lambda_V\right)$$

$$\left.\left.\times \exp\left[-\frac{\beta\epsilon}{2}\nu^2 - \beta\sum_{k=1}^c \phi(\nu - \nu_k)\right]\right\}^n\right\rangle_\Lambda^{1/n}.$$

$$\text{(A10)}$$

Λ_V is the capacity of the vertex fed by c trees $\mathbf{T}_1, \ldots, \mathbf{T}_c$.

Letting $y \equiv \nu - z$, we consider solutions of Eq. (12) in the form

$$R(z, \nu | \mathbf{T}) = W(\nu) Z_V(y | \mathbf{T}). \quad \text{(A11)}$$

Separating the dependence on the current potentials from that on the currents, the extra Gaussian distribution of ν in Eq. (A10) prevents the integration of ν from diverging. Indeed, in the $n \to 0$ limit and as $\epsilon \to 0$, the function $W(z)$ becomes independent of z and can be represented as

$$W(\nu) = \sqrt[4]{\frac{\beta\epsilon}{2\pi}}. \quad \text{(A12)}$$

The recursion relation involving the currents becomes decoupled to give

$$Z_V(y | \mathbf{T}) = \prod_{k=1}^{c-1} \left[\int dy_k Z_V(y_k | \mathbf{T}_k)\right]$$

$$\times \Theta\left(\sum_{k=1}^{c-1} y_k - y + \Lambda_{V(\mathbf{T})}\right) \exp\left[-\beta \sum_{k=1}^{c-1} \phi(y_k)\right]$$

$$\times \exp\left\{-\left\langle \ln\left\{\prod_{k=1}^c \left[\int dy_k Z_V(y_k | \mathbf{T}_k)\right]\right.\right.\right.$$

$$\left.\left.\left.\times \Theta\left(\sum_{k=1}^c y_k + \Lambda_V\right) \exp\left[-\beta \sum_{k=1}^c \phi(y_k)\right]\right\}\right\rangle_\Lambda\right\}.$$

$$\text{(A13)}$$

Let $F_V(y | \mathbf{T})$ be the vertex free energy when a current y is drawn from the vertex of a tree $\mathbf{T}$, given by $F_V(y | \mathbf{T}) = -T \ln Z_V(y | \mathbf{T})$. Then the recursion relation of the free energy is given by Eq. (13), which in the zero-temperature limit becomes Eq. (14).

To calculate the free energy in the replica approach, one returns to Eq. (7). In the second term of the exponential argument therein, one eliminates $\hat{Q}_{\mathbf{r,s}}$ by Eq. (A7), expresses $Q_{\mathbf{r,s}}$ in terms of $P_{\mathbf{s}}(\mathbf{z})$ by Eq. (9) and, in turn, $R(z_\alpha, \nu | \Lambda)$ by Eq. (11). In the third term, one expresses X in terms of $Q_{\mathbf{r,s}}$ by Eq. (A8) and follows similar steps. The result is

$$\langle Z^n \rangle = \exp\left(N\frac{c}{2} - \frac{c}{2} \left\langle \left\{ \int d\nu_1 d\nu_2 R(\nu_2, \nu_1 | \mathbf{T}_1) \right. \right. \right.$$

$$\left. \times R(\nu_1, \nu_2 | \mathbf{T}_2) \exp[-\beta\phi(\nu_1 - \nu_2)] \right\}^n \Big\rangle_\Lambda$$

$$+ \ln\left\langle \left\{ \left[\int d\nu \prod_{k=1}^{c} \left[\int d\nu_k R(\nu, \nu_k | \mathbf{T}_k) \right] \right. \right. \right.$$

$$\times \Theta\left(\sum_{k=1}^{c} \nu_k - c\nu + \Lambda \right)$$

$$\left. \left. \left. \times \exp\left(-\beta \sum_{k=1}^{c} \phi(\nu - \nu_k) - \frac{\beta\epsilon}{2}\nu^2 \right) \right\}^n \right\rangle_\Lambda \right).$$

$$(A14)$$

Using the recursion relation Eq. (12), one can show that the sum of the first two terms in the exponential argument vanishes. In the limit $n \to 0$ one obtains the free energy

$$\langle \beta F \rangle = -N \left\langle \ln\left\{ \int d\nu \prod_{k=1}^{c} [R(\nu, \nu_k | \mathbf{T}_k)] \Theta\left(\sum_{k=1}^{c} \nu_k - c\nu + \Lambda \right) \right. \right.$$

$$\left. \left. \times \exp\left[-\beta \sum_{k=1}^{c} \phi(\nu - \nu_k) - \frac{\beta\epsilon}{2}\nu^2 \right] \right\} \right\rangle_\Lambda. \quad (A15)$$

Using the vertex free energy representation, one then straightforwardly rewrites Eq. (A15) as Eq. (15) (up to a constant).

APPENDIX B: MESSAGES IN THE BAYESIAN APPROXIMATION

To show that the vertex free energies are directly related to passed messages in the Bayesian approximation, one resorts to formulating the problem on a bipartite graph and deriving the closed set of equations that relate to the messages passed from variables to interaction nodes and vice versa.

The representation of the problem as a bipartite graph is shown in Fig. 10, with the current variables y on the left and the interaction variables Z on the right. Using conventional notations [3] one can easily derive the closed set of equations:

$$Q(y_{ij}) \propto P(y_{ij})R(y_{ij}), \quad (B1)$$

where $Q(y_{ij})$ is the posterior of y_{ij} given Z_j, $P(y_{ij})$ is the prior of y_{ij}, and $R(y_{ij})$ is the likelihood of Z_j given y_{ij}. As shown in Fig. 10, the message from Z_j to y_{ij} is $Q(y_{ij})$, and the message from y_{ij} to Z_i is $R(y_{ij})$. Thus

$$R(y_{ij}) = \int \prod_{k \in \mathcal{L}_i \backslash j} dy_{jk} P(Z_j | y_{ij}, \{y_{jk} : k \in \mathcal{L}_j \backslash i\}) \prod_{k \in \mathcal{L}_j \backslash i} Q(y_{jk}).$$
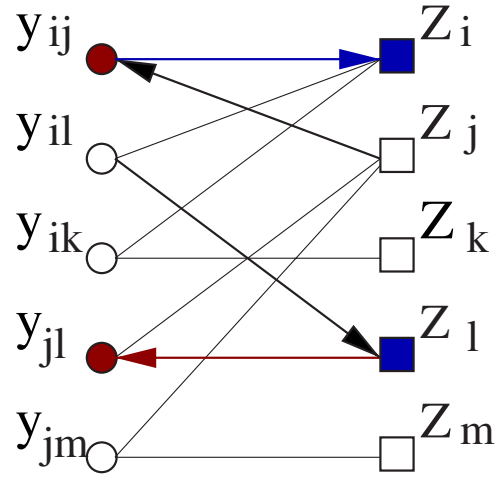
$$(B2)$$

Using

FIG. 10. (Color online) A bipartite graph representation of the resource allocation problem with the current variables y on the left and the interaction variables Z on the right.

$$P(y_{ij}) \propto \exp\left(-\frac{\beta}{2}\phi(y_{ij}) \right), \quad (B3)$$

$$P(Z_j | y_{ij}, \{y_{jk} : k \in \mathcal{L}_j \backslash i\}) \propto \Theta\left(\Lambda_j - y_{ij} + \sum_{k \in \mathcal{L}_j \backslash i} y_{jk} \right),$$

$$(B4)$$

and substituting the expression for Q (B1) into the P equation (B2) one obtains

$$R(y_{ij}) \propto \prod_{k \in \mathcal{L}_j \backslash i} \left(\int dy_{jk} \right) \Theta\left(\Lambda_j - y_{ij} + \sum_{k \in \mathcal{L}_j \backslash i} y_{jk} \right)$$

$$\times \prod_{k \in \mathcal{L}_j \backslash i} \exp\left(-\frac{\beta}{2}\phi(y_{jk}) \right) R(y_{jk}). \quad (B5)$$

Let $\tilde{F}_V(y_{ij}|Z_j) = -T \ln R(y_{ij})$. Then on taking the logarithm of both sides of Eq. (B5) and normalizing, one retrieves Eq. (15) if $\tilde{F}_V(y_{ij}|Z_j)$ is identified with the vertex free energy $F_V(y_{ij}|\mathbf{T}_j)$,

$$\tilde{F}_V(y_{ij}|Z_j) = -T \ln\left\{ \prod_{k=1}^{c-1} \left(\int dy_{jk} \right) \Theta\left(\sum_{k=1}^{c-1} y_{jk} - y_{ij} + \Lambda_j \right) \right.$$

$$\left. \times \exp\left[-\beta \sum_{k=1}^{c-1} (\tilde{F}_V(y_{jk}|Z_k) + \phi(y_{jk})) \right] \right\} - F_{av}.$$

$$(B6)$$

This means that the vertex free energy $F_V(y_{ij}|\mathbf{T}_j)$ is equivalent to $-T$ times the logarithm of the message $R(y_{ij})$ from y_{ij} to Z_i.

[1] H. Nishimori, *Statistical Physics of Spin Glasses and Information Processing* (Oxford University Press, Oxford, UK, 2001).

[2] M. Opper and D. Saad, *Advanced Mean Field Methods* (MIT, Cambridge, MA, 2001).

[3] D. J. C. Mackay, *Information Theory, Inference and Learning Algorithms* (Cambridge University Press, Cambridge, UK, 2003).

[4] J. S. Yedidia, W. T. Freeman, and Y. Weiss, IEEE Trans. Inf. Theory **51**, 2282 (2005).

[5] M. Mézard, e-print arXiv:cond-mat/0401237.

[6] M. Mézard and R. Zecchina, Phys. Rev. E **66**, 056126 (2002).

[7] Y. Kabashima and D. Saad, J. Phys. A **37**, R1 (2004).

[8] L. Saul and M. Jordan, Neural Comput. **6**, 1174 (1994).

[9] S. L. Lauritzen, J. Am. Stat. Assoc. **87**, 1098 (1992).

[10] N. S. Skantzos, I. P. Castillo, and J. P. L. Hatchett, Phys. Rev. E **72**, 066127 (2005).

[11] K. Y. M. Wong, D. Saad, and Z. Gao, *Advances in Neural Information Processing Systems*, edited by Y. Weiss, B. Schölkopf, and J. Platt (MIT, Cambridge, MA, 2005), Vol. 18, p. 1529.

[12] K. Y. Michael Wong and D. Saad, Phys. Rev. E **74**, 010104(R) (2006).

[13] D. Bertsekas, *Linear Network Optimization* (MIT, Cambridge, MA, 1991).

[14] L. Peterson and B. S. Davie, *Computer Networks: A Systems Approach* (Academic, San Diego, CA, 2000).

[15] Y. C. Ho, L. Servi, and R. Suri, Large Scale Syst. **1**, 51 (1980).

[16] S. Shenker, D. Clark, D. Estrin, and S. Herzog, Comput. Commun. Rev. **26**, 19 (1996).

[17] R. L. Rardin, *Optimization in Operations Research* (Prentice Hall, Englewood Cliffs, NJ, 1998).

[18] K. Y. M. Wong and D. Sherrington, J. Phys. A **20**, L793 (1987).

[19] This term is marginalized over all inputs to the current vertex, leaving the difference in current potentials y as its sole argument, hence the terminology used.

[20] If $\phi(y)$ is not an even function of y, analyses along the lines of Appendix A show that $\phi(y_k)$ in the Boltzmann factors of Eq. (13) have to be replaced by $\phi(\sigma_k y_k)$, where $\sigma_k = \pm 1$ should be quench averaged.

[21] Compared with the results in Fig. 2(a) (inset) of [12], here we have an even faster convergence to the high connectivity limit, after we have separated the contribution of the continuous component at $y=0$.

[22] F. P. Kelly, Eur. Trans Telecommun. Relat. Technol. **8**, 33 (1997).

[23] K. Y. M. Wong, C. H. Yeung, and D. Saad, *Neural Information Processing*, edited by I. King *et al.*, Lecture Notes in Computer Science Vol. 4233, Part II (Springer-Verlag, Berlin, 2006), p. 754.

[24] *On-line Learning in Neural Networks*, edited by D. Saad (Cambridge University Press, Cambridge, UK, 1998).