

Network pruning and growth: Probabilistic optimization

Yi-Zhi Xu  and David Saad 

The Nonlinearity and Complexity Research Group, Aston University, Birmingham B4 7ET, United Kingdom



(Received 24 November 2022; accepted 22 July 2023; published 8 August 2023)

Being the backbone of many human-made systems, networks require both pruning and growth to adapt to changing demand. We develop a message passing-based framework for analyzing and addressing the two-level optimization problem of edge removal/addition for indirectly dependent objectives. As exemplar problem we use routing in optical communication networks to minimize capability loss (removal) or maximize capacity (addition). The methods developed result in lower path lengths and higher capacity topologies with respect to existing ones and are suitable for a broad range of network design tasks.

DOI: [10.1103/PhysRevResearch.5.033087](https://doi.org/10.1103/PhysRevResearch.5.033087)

I. INTRODUCTION

Networks are prevalent in both natural and human-made structures, from electricity grids, road and human contact networks to wireless and optical communication networks. Network topology impacts directly and fundamentally on their performance, utilization, and functionality [1–4]. For instance, in the exemplar network, we focus on here, of optical communication, that underpins both access networks and the backbone of the Internet, network topology crucially impacts on transmission capacity, latency, resource requirements and cost [5–8]. Optimizing network topology is essential for better use of resource. One may consider adding or pruning edges for improving throughput or save resources with maximal or minimal impact on performance, respectively. The challenge is in the two-level optimization problem [9], where edge removal or addition hinges on an objective function which is optimized in parallel to the modified topology.

Network design clearly relies on the specific objective measure used that may be nontrivial to evaluate, since it requires a parallel optimization process. For instance, deletion or introduction of network edges in optical communication networks changes the network topology, and is evaluated by its impact on success measures that result from routing optimization given the modified topology, such as average path length, capacity or robustness. The joint-routing task of multiple communication requests is in itself generally NP-hard [10] (nondeterministic polynomial, where the computing time needed for obtaining solutions grows exponentially with the system size) and so is practical routing on wavelength-division-multiplexed optical networks that requires for paths to be node or edge disjoint, so that paths using similar wavelengths would not share nodes or edges [11–14]. Heuristics used for optical communication design focus on

vertex localities or distances between vertices [7,15,16], or on the introduction of other indirect objects that are easier to evaluate [8]. Most pruning/growth methods are greedy, considering one edge at a time, making the process scalable but result in suboptimal configurations.

The method we advocate using for network pruning and growth is belief propagation (BP), also known as message passing, which relates to the cavity method of statistical physics [17]. It has featured successfully in many network applications including both localized and nonlocalized interactions, e.g., multiwavelength node or edge-disjoint routing [14,18–20]. In the current task, two types of variables are introduced, one for routing optimization and one for edge removal/addition. Routing variables adhere to the path contiguity and node/edge-disjoint restrictions. BP can accommodate various constraints and objective functions, resulting in a closed set of equations, the solution of which is scalable and practical. In general BP and population dynamics also allows one to analyze generic properties of the system investigated. While the focus of this paper is optical communication networks, demonstrating the efficacy of the method to a particular application, a similar framework can be constructed for other network applications.

A practical way of designing a network is by making adjustments to an existing one, through pruning, adding or replacing edges. The framework we develop here revolves around edge removal since also the cases of edge addition and replacement make use of the deletion process of surplus added edges. The paper is organized as follows: to demonstrate the concept we first consider a simple routing scenario, where path length is the objective to be minimized and edges are removed using a BP-based approach; the framework is also used to evaluate the impact of judicious edge removal on network performance. This is followed by edge removal in a more realistic scenario of multiwavelength edge-disjoint routing, where different routes cannot share a common wavelength on the same edge (node disjoint, where routes cannot share a common wavelength on the same node is described in the Appendixes), and edge addition in the US optical backbone network CONUS60 [21].

Published by the American Physical Society under the terms of the Creative Commons Attribution 4.0 International license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI.

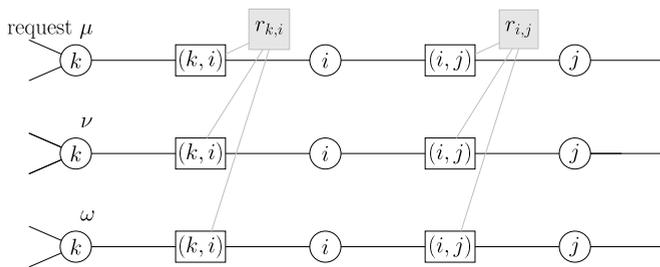


FIG. 1. Factor graph of the edge removal problem, where circle and rectangle nodes represent vertices and edges of the original network, respectively, and gray squares represent the edge removal factors.

II. EDGE REMOVAL AND AVERAGE PATH LENGTH

Consider a network G of single connected component with N vertices/nodes labeled i, j, k, \dots and E edges denoted as (i, j) , the task is to remove K edges such that the residual network will have the lowest objective function, average path length in the current case, for M arbitrary communication requests between uniformly and randomly chosen source-destination node pairs. The variable $\sigma_{i,j}^\mu$ on edge (i, j) for request μ , takes the value $\sigma_{i,j}^\mu = 1$ if the path for communication μ contains the directed edge from i to j , $\sigma_{i,j}^\mu = -1$ if it is from j to i , and $\sigma_{i,j}^\mu = 0$ otherwise [14]. Another variable $c_{i,j}$ represents the removal or retainment of edge (i, j) ($c_{i,j} = 0$ and $c_{i,j} = 1$, respectively). Note the interplay between the topology modification and optimization of the nonlocalized routing task, which is defined in this case by the Hamiltonian

$$H(\boldsymbol{\sigma}, \bar{c}) = \sum_{\mu} \sum_{(i,j)} |\sigma_{i,j}^\mu| + \lambda \sum_{(i,j)} c_{i,j}, \quad (1)$$

where $\boldsymbol{\sigma} \equiv [\dots, \bar{\sigma}^\mu, \dots] \equiv [\dots, (\dots, \sigma_{i,j}^\mu, \dots), \dots]$ is the complete routing configuration for all requests, \bar{c} the edge removal configuration, and the external field λ is adjusted to enforce the edge removal constraint [22,23] $\sum_{(i,j)} c_{i,j} = E - K$. Routing variables $\sigma_{i,j}^\mu$ in Eq. (1) are restricted by the routing continuity constraint, i.e., if neither vertex i nor j are the source/destination of request μ , there exists one and only one neighboring vertex k of i , denoted as $k \in \partial i$ where $\partial i \equiv \{j | (i, j) \in G\}$ is the nearest neighbors vertex set of i , such that $\sigma_{k,i}^\mu = \sigma_{i,j}^\mu$ where $\sigma_{i,j}^\mu \neq 0$, i.e., any incoming route exits exactly once. For source/destination node i only one neighboring variable $k \in \partial i$ admits $\sigma_{k,i}^\mu = \pm 1$ [14,19,24,25].

Belief propagation

Under the mean-field framework, the minimizing configuration of Eq. (1) is determined by the thermodynamic ground state as the temperature $T = 1/\beta \rightarrow 0$, where the probabilities of different configurations are described by the Boltzmann factor $e^{-\beta H(\boldsymbol{\sigma}, \bar{c})}$ [18]. To derive the BP equations we construct the factor graph shown in Fig. 1 where the state of edge (i, j) for request μ is directly influenced by the neighboring vertices i and j and removal factor node $r_{i,j}$.

While the factor graph exhibits short loops as interactions between different communications appear on every edge, the cross-request interaction is relatively weak with respect to

those along the entire path and contiguity constraint, and BP therefore results in good approximate solutions. More details are provided in the Appendixes. From the Bethe-Peierls approximation used in a broad range of applications [18,26–29], we obtain the BP equations

$$\begin{aligned} p_{i \rightarrow j}^\mu(0) &\propto \prod_{k \in \partial i \setminus j} q_{k \rightarrow i}^\mu(0) \\ &+ \sum_{m, n \in \partial i \setminus j} q_{m \rightarrow i}^\mu(1) q_{n \rightarrow i}^\mu(-1) \prod_{k \in \partial i \setminus \{j, m, n\}} q_{k \rightarrow i}^\mu(0), \\ p_{i \rightarrow j}^\mu(\pm 1) &\propto \sum_{k \in \partial i \setminus j} q_{k \rightarrow i}^\mu(\pm 1) \prod_{l \in \partial i \setminus j, k} q_{l \rightarrow i}^\mu(0), \\ q_{i \rightarrow j}^\mu(\sigma) &\propto e^{-\beta |\sigma|} r_{i,j}^\mu(\sigma) p_{i \rightarrow j}^\mu(\sigma), \\ r_{i,j}^\mu(0) &\propto \prod_{v \neq \mu} \tilde{r}_{i,j}^v(0) + e^{-\beta \lambda}, \quad r_{i,j}^\mu(\pm 1) \propto e^{-\beta \lambda}, \quad (2) \end{aligned}$$

where normalization terms are omitted for brevity, $\partial i \setminus j$ is the nearest neighbors vertex set of i except j , $p_{i \rightarrow j}^\mu(\sigma)$ is the cavity probability of edge (i, j) being in state σ with vertex i supporting request μ , and $q_{i \rightarrow j}^\mu(\sigma)$ is the probability of edge (i, j) being in state σ without the influence of vertex j for request μ ; finally, the factor message $r_{i,j}^\mu(\sigma)$ is the probability of $\sigma_{i,j}^\mu = \sigma$ due to the corresponding removal factor, after marginalization over $c_{i,j}$, linking routing and edge removal. The first term in the first line of Eq. (2) represents the case where communication μ does not pass through vertex i , and the second is where request μ uses two other edges for this transmission (m, i) (in) and (i, n) (out); in the second equation, communication μ uses (i, j) for entering/leaving vertex i , and another edge (k, i) for leaving/entering, respectively; the last equation includes the penalty $e^{-\beta}$ for using edge (i, j) by communication request μ . The message $\tilde{r}_{i,j}^\mu(\sigma) \propto e^{-\beta |\sigma|} p_{i \rightarrow j}^\mu(\sigma) p_{j \rightarrow i}^\mu(-\sigma)$ corresponds to the cavity probability of edge (i, j) being in state σ for request μ without the influence of the edge removal variable $c_{i,j}$; the marginal probability of each edge per request $q_{i,j}^\mu(\sigma) = \tilde{r}_{i,j}^\mu(\sigma)$, in the absence of edge removal [$\lambda = 0$ in Eq. (1)].

In many problems of this type, Eqs. (2) can be used, jointly with population dynamics, to obtain generic results for networks of given degree distribution and requests profile. However, the nonlocalized nature of the routing problem and the removal of specific edges that impacts on complete trajectories make this approach inappropriate for the current case. Instead, we average the results of Eq. (2) on instances with similar macroscopic properties to obtain generic results, alongside its use as the basis of an algorithm for specific cases.

After solving Eq. (2) numerically a number of times, we can calculate related quantities, such as the marginal probability $q_{i,j}^\mu(\sigma)$, edge removal probabilities $r_{i,j}(c)$ and the system entropy. The ground state average path length L/M (the BP curve in Fig. 4) is estimated from the entropy density $s(L/M)$, by averaging over inferred solutions of 20 samples per point as reported in the Appendixes. Using BP messages, we also developed a decimation algorithm, for removing one edge at a time, to obtain close-to-optimal removal configurations. Results shown in Fig. 2 demonstrate the performance of the algorithm in finding solutions close to the ground state, obtained by averaging 20 graph and request instances of similar

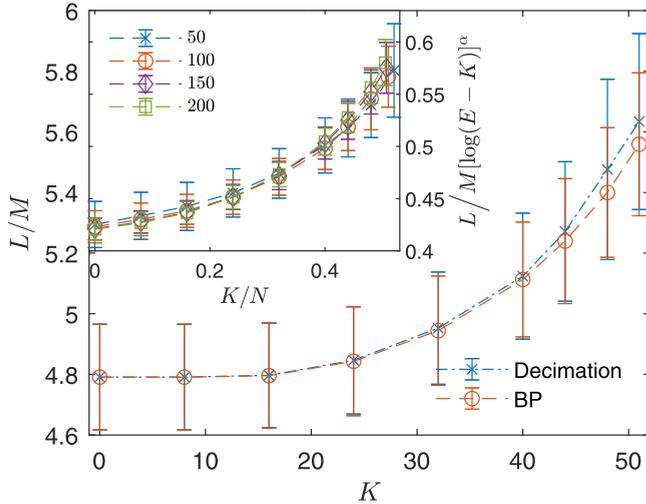


FIG. 2. BP predictions of ground state average path lengths, based on averaging different instances of similar characteristics, compared to results obtained by the decimation algorithm on random regular graph instances of degree 3, $N = 100$ vertices and $M = 100$ communications requests, where K edges are removed. Each point is based on 20 samples. (Inset) Scaling properties of average length due to edge removal in random regular graphs of degree 3, for networks of sizes $N = 50, 100, 150, 200$ and $M = N$ requests. The scaling parameter is $\alpha \approx 1.506$; each point is based on 20 samples. The logarithm in the figure is of base e .

characteristics. The inset of Fig. 2 shows the scaling properties of the method using results obtained for networks of sizes $N = 50, 100, 150, \text{ and } 200$. The average path length scales as $[\ln(E - K)]^{-\alpha}$ for different network sizes, which are all random-regular of degree 3, where the same fraction of edges are removed.

III. EDGE REMOVAL IN MULTIWAVELENGTH EDGE-DISJOINT ROUTING

Realistic optical communication networks utilize a high number of wavelengths simultaneously, where each wavelength on an edge cannot be used by two or more communication requests, but there is no interaction between transmissions using *different wavelengths*. The resulting marginal probability equation for edges is

$$q_{i,j}^{a,\mu}(\sigma) \propto r_{i,j}^{a,\mu}(\sigma) t_{i,j}^{a,\mu}(|\sigma|) p_{i \rightarrow j}^{a,\mu}(\sigma) p_{j \rightarrow i}^{a,\mu}(-\sigma), \quad (3)$$

where $r_{i,j}^{a,\mu}(\sigma)$ is the probability of assuming state σ due to the removal factor, as before; $t_{i,j}^{a,\mu}(|\sigma|)$ is the probability of assuming state σ due to the edge-disjoint constraint from using wavelength a on edge (i, j) to request μ .

To select the edges to be removed for both node and edge-disjoint routing, we use a reinforcement algorithm employing BP information, where a reinforcement factor $\eta_{i,j}$ is introduced, modifying the probabilities $q_{i,j}^{a,\mu}(\sigma) \leftarrow q_{i,j}^{a,\mu}(\sigma) \eta_{i,j}(|\sigma|)$ [20,23,30]. The reinforcement factor is updated as

$$\eta_{i,j}(\hat{c}) \leftarrow \varepsilon \eta_{i,j}(\hat{c}), \text{ where } \hat{c} = \arg \max_{c=0,1} r_{i,j}(c), \quad (4)$$

where ε is a growing variable, initialized with a value bigger than 1, and $r_{i,j}(0)$ is the edge removal probability.

The steps of the algorithm are as follows: (a) randomly initialize variables; (b) carry out BP iterations for a certain number of steps; (c) calculate the marginal probabilities $r_{i,j}(c)$; (d) construct the edge configuration \vec{c} where $c_{i,j} = \arg \max_{c=0,1} r_{i,j}(c)$, where $|\vec{c}| = E - K$; (e) if routing on the residual network allocates all given requests, output the edges of $c_{i,j} = 0$ and stop; and (f) otherwise, carry out more BP iterations and check the configuration. The multiwavelength node and edge-disjoint routing algorithms of [14] are used for routing allocation on the residual networks after edge removal.

Numerical results obtained for random regular networks of different sizes and removed edges are presented in Figs. 3(a)–3(c), 6, and Table I of the Appendixes, where BP guided reinforcement algorithms are compared with random edge removal (data used in the experiments can be found in Ref. [31]). To scale the number of requests to be allocated for systems of different sizes, we use the capacity (M_{\max}) of successfully allocated requests obtained by the corresponding algorithms [14] using $Q = 10$ wavelengths as shown in Figs. 3(a), 3(b), 6 and Table I. Using the scaled number of requests one obtains the average scaled path length for the various network sizes as a function of the removed edges as shown in Figs. 6 and 3(a). The reinforcement algorithm exhibits significantly shorter average scaled path length compared to random edge removal, and removal following the edge betweenness centrality order and reverse order [32], which have similar performance as random removal, as demonstrated in Fig. 7. In addition, the reinforcement algorithm can delete many more edges than random removal before they break down, under the same conditions, as shown in Fig. 3(b). Interestingly, our results show that edge removal does not influence the capacity (M_{\max}) significantly as shown in Fig. 3(c), where 10% of the edges are removed but the residual networks have capacities of 95% and 85% with respect to the node and edge-disjoint routing capacity prior to edge removal, respectively. The higher capacity drop when the same number of edges is removed in edge disjoint compared to node-disjoint routing, is due to the fact that capacity is originally higher in the former due to more intricate route choices and is therefore more sensitive to edge removal.

IV. NETWORK GROWTH IN MULTIWAVELENGTH EDGE-DISJOINT ROUTING

Another common scenario is adding new network edges to meet increasing demand. The methodology used for growth relies on a similar approach as that of network pruning. We focus on edge-disjoint routing since it is a more realistic scenario, although the same approach can also be used for node-disjoint routing. The method is based on adding a set S' of edges, and selecting $K \leq |S'|$ of them to be added to the original network.

The method hinges on the choice of the edge set S' from all available edges outside the original edge set of size E [examining the finite size edge set S' is computationally tractable compared to considering $O(N^2)$ potential edges], by examining the impact of individual additions on reducing routing costs (e.g., average path length). Once the set S' has been selected, K edges are chosen from it to be added in one of

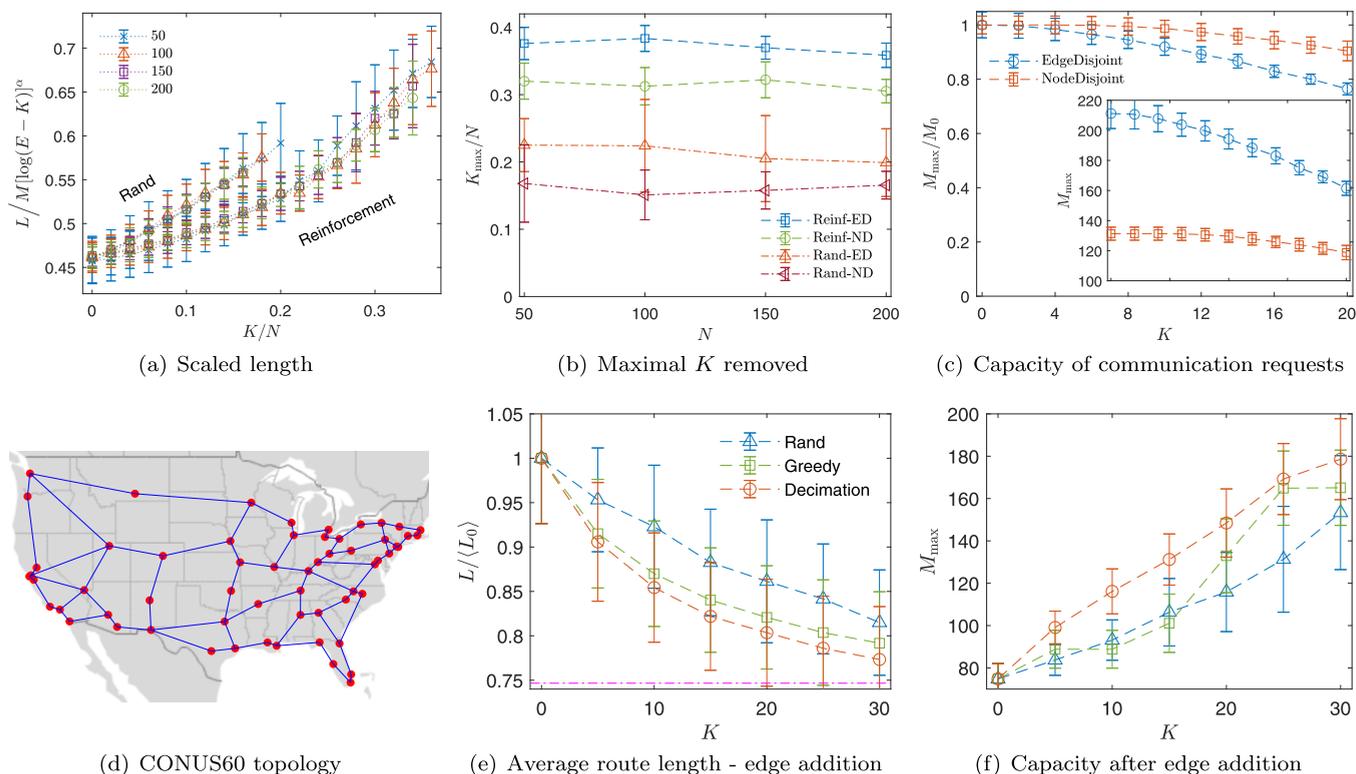


FIG. 3. (a) Performance of reinforcement-based and random edge removal algorithms under edge-disjoint routing for networks of size N . The scaling coefficient for random removal is $\alpha = 1.4539$ and for the reinforcement-based algorithm $\alpha = 1.4514$. Each point is based on at least 15 samples. The logarithm in the figure is of base e . (b) Maximal number of edges that can be removed by the reinforcement (Reinf) and random (Rand) algorithms under node-disjoint (ND) and edge-disjoint (ED) routing on random regular networks of degree 3 and of different sizes N . Each point is based on 20 samples. (c) With K edges removed, the capacities of random regular networks decrease slowly. Results were obtained for networks with degree 3 and size $N = 100$, and the number of wavelength $Q = 10$. Each point is averaged on 20 instances. (d) Network growth by adding K edges to CONUS60 with geographic distances as edge weights [21]. (e) the relative average path length after K edges were added with respect to that of the original graph $\langle L_0 \rangle$. (f) the capacities when K edges introduced. The bottom magenta dash-dotted line is the relative total path length of adding all the 60 edges in the external edge set, which is the lower bound. Both (e) and (f) are under the edge-disjoint scenario, the number of wavelengths is $Q = 11$ and the number of requests is $M = N = 60$. Each point is averaged over 20 instances.

three methods: (i) random—where K edges in S' are selected at random; (ii) greedy selection—where K edges with the highest impact are selected; (iii) BP decimation—where K edges with the largest $r_{i,j}(1)$ values (after BP iterations are performed) are set to state 1 (connected).

We test the three algorithms on the 60-node Continental United States (CONUS60) optical communication network [21] under the same edge set S' for a fair comparison. The original network has 79 edges, and $M = N = 60$ random requests are to be allocated on $Q = 11$ wavelengths. The external edge set S' is of size 60 is obtained by a greedy

algorithm. Figs. 3(d)–3(f) show decimation is more effective than random or greedy allocation, in finding configurations that increase capacities and decrease average path length the most.

V. DISCUSSION

Network topology design is challenging due to the optimization of edge deletion/addition to extremize an objective that is in itself obtained through nontrivial optimization. While graph properties are often used for the design of networks, they typically ignore the specific task objectives. The BP-guided decimation algorithms developed here provide better solutions than existing alternatives, with lower path lengths and higher capacities. The average path length is chosen here as the objective function; however, the framework can be generalized to accommodate other nonlinear costs and edge weights [14]. As all BP-based approaches, also the current method is exact only on trees. Results are likely to deteriorate as the number of short loops increases, but since practical networks operate away from criticality we expect good performance in a broad range of operational conditions. While the exemplar task we selected is of great importance

TABLE I. The number of requests (M) allocated to $Q = 10$ wavelengths on random regular networks of degree 3 and different sizes (N). The value of M selected for each of the cases is proportional to the capacity (M_{\max}) obtained by the corresponding routing algorithms of Ref. [14].

N	50	100	150	200
$M_{\text{Node Disjoint}}$	64	100	132	162
$M_{\text{Edge Disjoint}}$	67	100	129	155

for optical communication networks that carry most of the Internet traffic, network design has significant impact on many other applications, from the design and maintenance prioritization of road networks given demand [33], the design of VLSI circuits [34,35] and reducing electricity networks [36,37] to the pruning of metabolic networks based on flux [38]. This framework can be instrumental for addressing these and many other real-world applications. It can also help simplify approximation tasks by reducing the number of variable interactions/constraints in a manner that minimizes impact on the original target Hamiltonian.

ACKNOWLEDGMENT

D.S. and Y.-Z.X. acknowledge support from the Engineering and Physical Sciences Research Council (EPSRC) Programme Grant TRANSNET (EP/R035342/1).

APPENDIX A: EDGE REMOVAL AND AVERAGE PATH LENGTH

The motivation for removing/adding edges is different from one application to another. In optical communication networks, maintaining optical cables is very costly and the industry looks for edges to be removed to save costs, similarly, they would like to invest prudently in laying new cables; all of this is conditioned on traffic and transmission demands. In VLSI design, removing unnecessary edges reduces cross-noise and interference, allowing for more components to be used per area. In complex metabolic network, pruning nonessential links helps in identifying the underlying mechanism, which is not always fully known, ignoring less relevant paths.

In the energy function of the system, Eq. (1), the edge removal state $c_{i,j}$ restricts the routing variables $\sigma_{i,j}^\mu$ on edge (i, j) , since if the edge is removed there would be no transmissions through it. This restriction is expressed by the conditioning of the variable not being zero on the existence of an edge

$$\sigma_{i,j}^\mu \leftarrow c_{i,j} \sigma_{i,j}^\mu, \quad \forall \mu. \quad (\text{A1})$$

The factor graph of Fig. 1 of the main text represents the interactions between vertices, edges and removal constraints.

From the energy function Eq. (1) and considering all the constraints, we obtain the partition function

$$Z(\beta) = \sum_{\sigma, \vec{c}} \prod_{(i,j)} e^{-\beta \lambda c_{i,j}} \cdot \prod_{(i,j), \mu} e^{-\beta |\sigma_{i,j}^\mu|} \delta[\sigma_{i,j}^\mu, c_{i,j} \sigma_{i,j}^\mu] \cdot \prod_{i, \mu} \delta \left[\sum_{j \in \partial i} \sigma_{j,i}^\mu, -A_i^\mu \right] \Theta \left[2 - \sum_{k \in \partial i} |\sigma_{k,i}^\mu| \right], \quad (\text{A2})$$

where the inverse temperature β is introduced, $\partial i \equiv \{j | (i, j) \in G\}$ is the nearest-neighbor vertex set, $\delta[x, y]$ is the Kronecker delta function, $\delta[x, y] = 1$ if $x = y$ and 0 otherwise, and $\Theta[x]$ is the Heaviside step function, $\Theta[x \geq 0] = 1$ and 0 otherwise. There are three terms in the partition function: the first represents the edge removal cost, the second is the path length objective and removal implications of Eq. (A1), and the last term is the continuity constraint from

vertex to neighboring edge [e.g., vertex i to edge (i, j) and (k, i) for each request in the factor graph of Fig. 1]. The variable A_i^μ in Eq. (A2) provides source/destination information for request μ , if vertex i is the source/destination of request μ : $A_i^\mu = 1/-1$, and $A_i^\mu = 0$ otherwise.

1. Belief-propagation equations

A natural question is the reason behind the demonstrated success of the algorithm given the loopy structure of the factor graphs. In the case of linear cost with either NDP (node-disjoint paths) or EDP (edge-disjoint paths) constraints, different wavelength layers interact only through the sources and destinations (vertex i —where $A_i^\mu = \pm 1$ for request μ), and if the original graph is acyclic we expect only loops of larger size to exist. In more general cases with nonlinear costs on edges, interactions between layers are found on every edge [through a general cost function $F_{i,j}(n)$, where n is the number of wavelengths used in edge (i, j)], and hence the resulting graphs are loopy for both NDP and EDP scenarios. Nevertheless, the loopy cross-wavelength interaction is relatively weak with respect to the interaction along the entire lightpath and the contiguity constraint. More specifically, in the EDP/NDP cases the cross-layer interaction is independent of the specific wavelength assignment as it only depends on the number of wavelengths used per edge, while messages within layers are a results of a complete trajectory of interaction between nodes/edges, verifying contiguity, wavelength availability and edge load [14].

Following the Bethe-Peierls approximation [26,27], the marginal probability $q_{i,j}^\mu(\sigma)$ of edge (i, j) for request μ is calculated by

$$q_{i,j}^\mu(0) = \frac{1}{z_{i,j}^\mu} r_{i,j}^\mu(0) p_{i \rightarrow j}^\mu(0) p_{j \rightarrow i}^\mu(0),$$

$$q_{i,j}^\mu(\pm 1) = \frac{1}{z_{i,j}^\mu} e^{-\beta} r_{i,j}^\mu(\pm 1) p_{i \rightarrow j}^\mu(\pm 1) p_{j \rightarrow i}^\mu(\mp 1), \quad (\text{A3})$$

where $z_{i,j}^\mu$ is the normalization factor

$$z_{i,j}^\mu = \sum_{\sigma} e^{-\beta |\sigma|} r_{i,j}^\mu(\sigma) p_{i \rightarrow j}^\mu(\sigma) p_{j \rightarrow i}^\mu(-\sigma). \quad (\text{A4})$$

The edge variable $\sigma_{i,j}^\mu$ in the factor graph of Fig. 1 is influenced by neighboring nodes, vertices i and j and constraint factor $r_{i,j}$, through the messages $p_{i \rightarrow j}^\mu(\sigma)$, the cavity probability of edge (i, j) being in state σ with vertex i supporting request μ , $p_{j \rightarrow i}^\mu(\sigma)$ and $r_{i,j}^\mu(\sigma)$, the probability of $\sigma_{i,j}^\mu = \sigma$ after marginalization over $c_{i,j}$. Another cavity probability to be defined is $q_{i \rightarrow j}^\mu(\sigma)$, being the probability of edge (i, j) being in state σ without the influence of vertex j for request μ .

If $A_i^\mu = 0$, the routing continuity constraint takes the form

$$z_i^\mu = \prod_{j \in \partial i} q_{j \rightarrow i}^\mu(0) + \sum_{j, k \in \partial i} q_{j \rightarrow i}^\mu(1) q_{k \rightarrow i}^\mu(-1) \prod_{l \in \partial i \setminus \{j, k\}} q_{l \rightarrow i}^\mu(0), \quad (\text{A5})$$

where the first term represents the case where vertex i is not a part of path μ , and the second that there exists a two edges construct $j \rightarrow i \rightarrow k$ as a part of the path for request μ . If $A_i^\mu \neq 0$, vertex i is a source or destination of request μ ,

making the continuity constraint

$$z_i^\mu = \sum_{j \in \partial i} q_{j \rightarrow i}^\mu (-A_i^\mu) \prod_{k \in \partial i \setminus j} q_{k \rightarrow i}^\mu(0), \quad (\text{A6})$$

where there exists one edge (i, j) for accommodating the outgoing/incoming path from the source/destination vertex.

In the partition function of Eq. (A2), remaining edges contribute $e^{-\beta\lambda}$ each, totaling

$$z_{i,j} = \prod_{\mu} \tilde{r}_{i,j}^\mu(0) + e^{-\beta\lambda} \prod_{\mu} \sum_{\sigma} \tilde{r}_{i,j}^\mu(\sigma), \quad (\text{A7})$$

where $\tilde{r}_{i,j}^\mu(\sigma) \propto e^{-\beta|\sigma|} p_{i \rightarrow j}^\mu(\sigma) p_{j \rightarrow i}^\mu(-\sigma)$ is the cavity probability of edge (i, j) for request μ without the influence of the edge removal state $c_{i,j}$. The removal probabilities are calculated by

$$\begin{aligned} r_{i,j}(0) &= \frac{1}{z_{i,j}} \prod_{\mu} \tilde{r}_{i,j}^\mu(0), \\ r_{i,j}(1) &= \frac{1}{z_{i,j}} e^{-\beta\lambda} \prod_{\mu} \sum_{\sigma} \tilde{r}_{i,j}^\mu(\sigma) = \frac{1}{z_{i,j}} e^{-\beta\lambda}, \end{aligned} \quad (\text{A8})$$

where the equality in the second equation is based on the normalization $\sum_{\sigma} \tilde{r}_{i,j}^\mu(\sigma) = 1$. From the constraint in Eq. (A1), $c_{i,j} = 0$ leads to $\sigma_{i,j}^\mu = 0, \forall \mu$ and consequently $\prod_{\mu} \tilde{r}_{i,j}^\mu(0)$ in the first equation; while $c_{i,j} = 1, \sigma_{i,j}^\mu$ leads to the more general product $\prod_{\mu} \sum_{\sigma} \tilde{r}_{i,j}^\mu(\sigma)$ in the second.

The corresponding BP equations (2) for $A_i^\mu = 0$ are

$$\begin{aligned} p_{i \rightarrow j}^\mu(0) &\propto \prod_{k \in \partial i \setminus j} q_{k \rightarrow i}^\mu(0) \\ &+ \sum_{m,n \in \partial i \setminus j} q_{m \rightarrow i}^\mu(1) q_{n \rightarrow i}^\mu(-1) \prod_{k \in \partial i \setminus j, m, n} q_{k \rightarrow i}^\mu(0), \\ p_{i \rightarrow j}^\mu(\pm 1) &\propto \sum_{k \in \partial i \setminus j} q_{k \rightarrow i}^\mu(\pm 1) \prod_{l \in \partial i \setminus j, k} q_{l \rightarrow i}^\mu(0); \\ q_{i \rightarrow j}^\mu(\sigma) &\propto e^{-\beta|\sigma|} r_{i,j}^\mu(\sigma) p_{i \rightarrow j}^\mu(\sigma), \end{aligned} \quad (\text{A9})$$

and for $A_i^\mu \neq 0$

$$\begin{aligned} p_{i \rightarrow j}^\mu(0) &\propto \sum_{k \in \partial i \setminus j} q_{k \rightarrow i}^\mu(-A_i^\mu) \prod_{l \in \partial i \setminus j, k} q_{l \rightarrow i}^\mu(0), \\ p_{i \rightarrow j}^\mu(A_i^\mu) &\propto \prod_{k \in \partial i \setminus j} q_{k \rightarrow i}^\mu(0), \\ p_{i \rightarrow j}^\mu(-A_i^\mu) &= 0. \end{aligned} \quad (\text{A10})$$

The influence of the removal constraint $r_{i,j}^\mu(\sigma)$ in Eq. (2) is computed by

$$\begin{aligned} r_{i,j}^\mu(0) &\propto \prod_{v \neq \mu} \tilde{r}_{i,j}^v(0) + e^{-\beta\lambda} \prod_{v \neq \mu} \sum_{\sigma} \tilde{r}_{i,j}^v(\sigma) \\ &= \prod_{v \neq \mu} \tilde{r}_{i,j}^v(0) + e^{-\beta\lambda}, \\ r_{i,j}^\mu(\pm 1) &\propto e^{-\beta\lambda} \prod_{v \neq \mu} \sum_{\sigma} \tilde{r}_{i,j}^v(\sigma) = e^{-\beta\lambda}. \end{aligned} \quad (\text{A11})$$

Note that the arguments in $\tilde{r}_{i,j}^\mu(\sigma)$ in Eq. (A11) and $r_{i,j}(c)$ in Eq. (A8) are different, where $\sigma \in \{0, \pm 1\}$ and

$c \in \{0, 1\}$, the routing state and removal state variables, respectively.

To calculate the BP equations, we obtain the value of the external field λ , which is defined by $\sum_{(i,j)} c_{i,j} = E - K$ and, in mean-field framework, can be written as

$$E - K = \sum_{(i,j)} r_{i,j}(1) = \sum_{(i,j)} \frac{e^{-\beta\lambda}}{\tilde{r}_{i,j} + e^{-\beta\lambda}}, \quad (\text{A12})$$

where $\tilde{r}_{i,j} = \frac{\prod_{\mu} \tilde{r}_{i,j}^\mu(0)}{\prod_{\mu} \sum_{\sigma} \tilde{r}_{i,j}^\mu(\sigma)} = \prod_{\mu} \tilde{r}_{i,j}^\mu(0)$. As $\tilde{r}_{i,j} \geq 0$, the second equation is monotonically increasing as a function of $e^{-\beta\lambda}$, and therefore can be easily solved numerically using the bisection or fixed point iteration methods.

We iterate the BP equations (or part of them) synchronously and calculate λ using the cavity probabilities ($p_{\rightarrow}^\mu, q_{\rightarrow}^\mu$ and $r_{i,j}^\mu$) until convergence or until a predetermined iteration limit is reached. The corresponding marginal probabilities and external field can then be calculated as well as the free energy, entropy, and average path length.

2. Replica symmetric mean-field theory

Using the relation $F = -\frac{1}{\beta} \ln Z$ [18,28], we calculate the factors and nodes contributing to the free energy

$$\begin{aligned} F &= \sum_{\mu, (i,j)} f_{i,j}^\mu + \sum_{\mu, i} f_i^\mu + \sum_{(i,j)} f_{i,j} \\ &- \sum_{\mu, (i,j)} (f_{i,j}^{\mu,i} + f_{i,j}^{\mu,j} + f_{i,j}^{\mu,\mu}), \end{aligned} \quad (\text{A13})$$

where $f_{i,j}^\mu = -\frac{1}{\beta} \ln z_{i,j}^\mu$, $f_i^\mu = -\frac{1}{\beta} \ln z_i^\mu$ and $f_{i,j} = -\frac{1}{\beta} \ln z_{i,j}$, which are free energy contributions from the various factor and variable nodes: edge-request factors (rectangle node (i, j) of request μ in Fig. 1), vertex-request nodes (circle node i of request μ) and edge removal constraint factors (e.g., gray square $r_{i,j}$) and their neighboring links respectively. Additionally, the three last terms are free energy contributions from links in the factor graph, which have been considered twice and should be subtracted. The resulting equations are

$$\begin{aligned} f_{i,j}^{\mu,i} &= -\frac{1}{\beta} \ln \left[\sum_{\sigma} p_{i \rightarrow j}^\mu(\sigma) q_{j \rightarrow i}^\mu(-\sigma) \right], \\ f_{i,j}^{\mu,j} &= -\frac{1}{\beta} \ln \left[\sum_{\sigma} p_{j \rightarrow i}^\mu(\sigma) q_{i \rightarrow j}^\mu(-\sigma) \right], \\ f_{i,j}^{\mu,\mu} &= -\frac{1}{\beta} \ln \left[\sum_{\sigma} \tilde{r}_{i,j}^\mu(\sigma) r_{i,j}^\mu(\sigma) \right]. \end{aligned} \quad (\text{A14})$$

The energy function for the system is

$$\begin{aligned} H &= \sum_{\mu, (i,j)} [1 - q_{i,j}^\mu(0)] + \lambda \sum_{(i,j)} r_{i,j}(1) \\ &= L + \lambda(E - K), \end{aligned} \quad (\text{A15})$$

where L is the total path length.

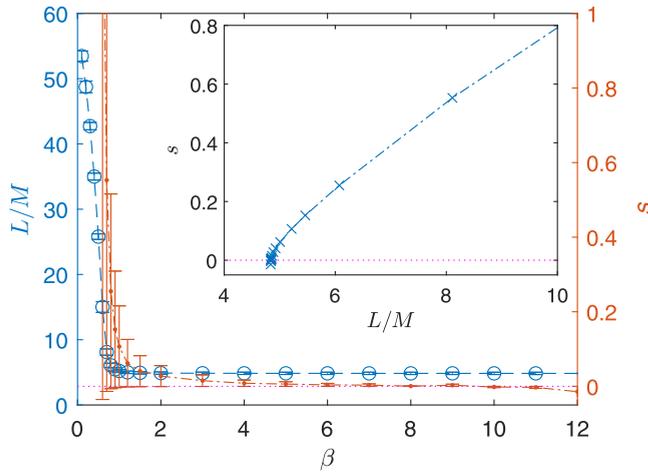


FIG. 4. Belief-propagation simulations on 100-node random regular graph of degree 3 and 100 requests, where 24 edges are to be removed. The inset focuses on the ground-state value of L/M where $s = 0$. Each point is averaged over 20 instances.

The resulting entropy

$$S = \beta(H - F) \quad (\text{A16})$$

is related to the number of solutions, and the entropy density is defined as $s \equiv S/(MN)$.

The numerical entropy density average given path lengths is used to estimate the average path lengths of the ground state. There are two different situations: when zero-entropy states are reached, the ground state is identified at $s = 0$ and the corresponding value of L/M constitute the theoretical optimal average path length, e.g., in Fig. 4 $\langle L/M \rangle_{\min} \approx 4.843$, according to the inset; if L/M and s remain flat and positive as β increases, this value of L/M is the ground state estimate (this may be a property of the network topology, which offers multiple shortest paths). In Fig. 2, we show the ground state predictions in random regular networks of degree 3 and size 100, based on 20 randomly generated specific networks and request sets. We average over the instances for each β , to obtain the curves of L/M and s with standard deviation represented by error bars in Fig. 4. These results give rise to the dependence of L/M on K in Fig. 2 of the main text. In the flat L/M and s regimes of Fig. 4 (that correspond to small K values in Fig. 2), the resulting error bars are with very small giving rise to the corresponding ground state average path length $\langle L/M \rangle$ and error bars of Fig. 2. To find the ground states we look for zero-entropy L/M values, where the optimal average path length is estimated by fitting the points of $(\langle L/M \rangle, \langle s \rangle)$ near $s = 0$ (inset of Fig. 4); the error bars are the standard deviation of the nearest point around $s = 0$.

3. Decimation algorithm

BP messages and marginal probabilities give rise to a guided decimation algorithm for edge removal. The steps of the algorithm are

0. input the network, set of communication requests and the edge removal number K , set β to a large value and initialize the messages $\{p_{\rightarrow}^{\mu}\}$ randomly;

1. calculate $\{\tilde{r}_{i,j}^{\mu}\}$, update λ , $\{r_{i,j}^{\mu}\}$ and $\{q_{\rightarrow}^{\mu}\}$, then update $\{p_{\rightarrow}^{\mu}\}$ according to the related equations (2) and (A9)–(A12);
2. repeat step 1 a predefined number of times or until convergence, then calculate the edge removal probabilities $\{r_{i,j}(0)\}$; fix the state $c_{i,j} = 0$ for the edge with the largest removal probability $r_{i,j}(0)$ and remove the edge to simplify the network;
3. repeat step 2 K times, output the residual network.

The performance of the decimation algorithm is shown in Fig. 2, which is close to the ground state described by the mean-field theory. More numerical results on random regular networks of different sizes $N = 50, 100, 150$ and 200 are shown in the inset of Fig. 2.

To obtain the collapse fitting parameter α in Figs. 2, 3(a), and 6, we first select data for given (N, L) and the same value of K/N , then use logarithmic scaling $(\ln \ln(E - K), \ln L)$ to obtain the first order coefficient $a(K/N)$ by linear fitting; α is the average value of $a(K/N)$ over different K/N ratios. The general logarithmic scaling is assumed due to the expected path lengths scaling logarithmically with the number of edges; the power α encapsulate the fine differences between the emerging structures after edge removal.

APPENDIX B: EDGE REMOVAL IN NODE-DISJOINT ROUTING

In the multiwavelength node-disjoint routing scenario, wavelengths used on a node (vertex) cannot be shared by different communication paths; there is no interaction between the transmissions using different wavelengths on the same node/edge. Compared with the general routing problem, this scenario has two additional constraints—the node disjoint constraint and a restriction on the choice of wavelength, where each communication request uses one and only one wavelength. The marginal vertex and edge probabilities for each wavelength and communication request are given by

$$\begin{aligned} p_i^{a,\mu}(0) &\propto t_i^{a,\mu}(0) \prod_{j \in \partial i} q_{j \rightarrow i}^{a,\mu}(0), \\ p_i^{a,\mu}(1) &\propto t_i^{a,\mu}(1) \sum_{m,n \in \partial i} q_{m \rightarrow i}^{a,\mu}(1) q_{n \rightarrow i}^{a,\mu}(-1) \prod_{j \in \partial i \setminus \{m,n\}} q_{j \rightarrow i}^{a,\mu}(0), \\ q_{i,j}^{a,\mu}(\sigma) &\propto r_{i,j}^{a,\mu}(\sigma) p_{i \rightarrow j}^{a,\mu}(\sigma) p_{j \rightarrow i}^{a,\mu}(-\sigma), \end{aligned} \quad (\text{B1})$$

where $t_i^{a,\mu}$ is the node disjoint constraint from using wavelength a on vertex i for request μ , and $r_{i,j}^{a,\mu}$ the influence of edge removal on the state of edge (i, j) using wavelength a for request μ .

In multiwavelength routing, edge removal $c_{i,j} = 0$ influences routing variables on all wavelengths and all requests since $\sigma_{i,j}^{a,\mu} \leftarrow c_{i,j} \sigma_{i,j}^{a,\mu}, \forall \mu, a$, hence the marginal removal probability should be calculated by

$$\begin{aligned} r_{i,j}(0) &\propto \prod_{a,\mu} \tilde{r}_{i,j}^{a,\mu}(0), \\ r_{i,j}(1) &\propto e^{-\beta \lambda} \prod_{a,\mu} \sum_{\sigma} \tilde{r}_{i,j}^{a,\mu}(\sigma), \end{aligned} \quad (\text{B2})$$

where $\tilde{r}_{i,j}^{a,\mu}(\sigma) = p_{i \rightarrow j}^{a,\mu}(\sigma) p_{j \rightarrow i}^{a,\mu}(-\sigma)$. The message from removal factor node to wavelength a on edge (i, j) of request

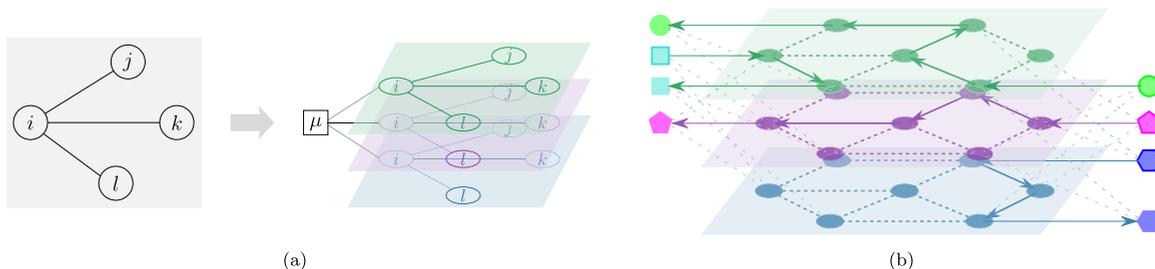


FIG. 5. (a) Mapping the original network (left) onto multilayer replica networks that use different wavelengths (right). In this example, node i is the source/destination of request μ . Introducing an auxiliary node μ , denoted by a square and connected to nodes i at each of the layers, facilitates message passing between the new node and the different layers to determine the allocation of transmissions to wavelengths. These auxiliary nodes also facilitate the interaction among different wavelengths. (b) An example of routing paths on a simple network with $N = 7$ nodes and $M = 4$ requests, which shows the complete algorithmic framework with wavelengths as colored layers, sources and destinations as auxiliary nodes denoted by elliptic symbols on the planes and nonelliptic symbols, respectively; transmission paths are represented by colored solid lines with arrows and unoccupied wavelength channels as colored dotted lines.

μ is computed by

$$r_{i,j}^{a,\mu}(0) \propto \prod_{(b,v) \neq (a,\mu)} \tilde{r}_{i,j}^{b,v}(0) + e^{-\beta\lambda} \prod_{(b,v) \neq (a,\mu)} \sum_{\sigma} \tilde{r}_{i,j}^{b,v}(\sigma),$$

$$r_{i,j}^{a,\mu}(\pm 1) \propto e^{-\beta\lambda} \prod_{(b,v) \neq (a,\mu)} \sum_{\sigma} \tilde{r}_{i,j}^{b,v}(\sigma). \quad (\text{B3})$$

To enforce the node-disjoint constraint, where at most one request can be transmitted through each wavelength on an edge, the messages from the corresponding factor to each node are computed by

$$t_i^{a,\mu}(0) \propto \prod_{v \neq \mu} \tilde{t}_i^{a,v}(0) + e^{-\beta w_i} \sum_{v \neq \mu} \tilde{t}_i^{a,v}(1) \prod_{\omega \neq \mu, v} \tilde{t}_i^{a,\omega}(0),$$

$$t_i^{a,\mu}(1) \propto e^{-\beta w_i} \prod_{v \neq \mu} \tilde{t}_i^{a,v}(0). \quad (\text{B4})$$

The first term in the first equation of (B4) is the case where no communication path uses wavelength a on vertex i ; the second is where communication path v takes wavelength a on vertex i , therefore this wavelength cannot be used by other communication paths on this vertex. Passing through vertex i , also incurs a node cost w_i that may be arbitrarily defined for the specific problem at hand. The second equation corresponds to communication path μ using wavelength a on vertex i , therefore, it cannot be used by other communication paths; this also incurs a cost of w_i . Notice that while using a vertex incurs some cost using edges does not (reversed in edge-disjoint routing). The auxiliary probabilities are defined as

$$\tilde{t}_i^{a,\mu}(0) \propto \prod_{j \in \partial i} q_{j \rightarrow i}^{a,\mu}(0),$$

$$\tilde{t}_i^{a,\mu}(1) \propto \sum_{j,k \in \partial i} q_{j \rightarrow i}^{a,\mu}(1) q_{k \rightarrow i}^{a,\mu}(-1) \prod_{l \in \partial i \setminus j,k} q_{l \rightarrow i}^{a,\mu}(0), \quad (\text{B5})$$

being proportional to the probabilities of wavelength a on vertex i being used by communication μ (or not), respectively, in the absence of the node-disjoint constraint.

Belief-propagation iteration with reinforcement

Considering all the quantities we need, the belief-propagation equations become

$$p_{i \rightarrow j}^{a,\mu}(0) \propto t_i^{a,\mu}(0) \prod_{k \in \partial i \setminus j} q_{k \rightarrow i}^{a,\mu}(0) + t_i^{a,\mu}(1)$$

$$\times \sum_{m,n \in \partial i \setminus j} q_{m \rightarrow i}^{a,\mu}(1) q_{n \rightarrow i}^{a,\mu}(-1) \prod_{k \in \partial i \setminus j, m, n} q_{k \rightarrow i}^{a,\mu}(0),$$

$$p_{i \rightarrow j}^{a,\mu}(\pm 1) \propto t_i^{a,\mu}(1) \sum_{k \in \partial i \setminus j} q_{k \rightarrow i}^{a,\mu}(\pm 1) \prod_{l \in \partial i \setminus j, k} q_{l \rightarrow i}^{a,\mu}(0);$$

$$q_{i \rightarrow j}^{a,\mu}(\sigma) \propto r_{i,j}^{a,\mu}(\sigma) p_{i \rightarrow j}^{a,\mu}(\sigma). \quad (\text{B6})$$

The same approach is being used when it comes to source/destination vertices. However, we add two external auxiliary vertices for each request, connected to each source or destination vertex, for assigning particular wavelengths for each request [14]. The messages from auxiliary vertices are

$$q_{\mu \rightarrow i}^a(0) \propto e^{-\beta} \sum_{b \neq a} q_{i \rightarrow \mu}^b(-A_i^\mu) \prod_{c \neq a, b} q_{i \rightarrow \mu}^c(0),$$

$$q_{\mu \rightarrow i}^a(A_i^\mu) \propto e^{-\beta} \prod_{b \neq a} q_{i \rightarrow \mu}^b(0),$$

$$q_{\mu \rightarrow i}^a(-A_i^\mu) = 0, \quad (\text{B7})$$

where the symbol “ μ ” in the subscripts “ $i \rightarrow \mu$ ” and “ $\mu \rightarrow i$ ” stands for the auxiliary source and destination vertices of request μ as described in Figs. 5(a) and 5(b).

The algorithm with reinforcement includes the following steps

0. read the graph and request set, input the value of the number of edges to be removed K and wavelengths to be used Q , set the value β , and randomly initialize $\{q_{i \rightarrow}^\mu\}$;
1. using the value of $\{q_{i \rightarrow}^\mu\}$, calculate $\{t_i^{a,\mu}\}$ and $\{p_{i \rightarrow}^\mu\}$;
2. using $\{p_{i \rightarrow}^\mu\}$, calculate $\{\tilde{r}_{i,j}^{a,\mu}\}$, then λ and $\{r_{i,j}^{a,\mu}\}$;
3. according to $\{r_{i,j}^{a,\mu}\}$ and $\{p_{i \rightarrow}^\mu\}$, compute $\{q_{i \rightarrow}^\mu\}$.

Repeating the steps 1–3 we calculate the related probabilities and quantities of interest.

The reinforcement factor $\eta_{i,j}$ is introduced to accelerate the process of determining the edge removal marginal probabili-

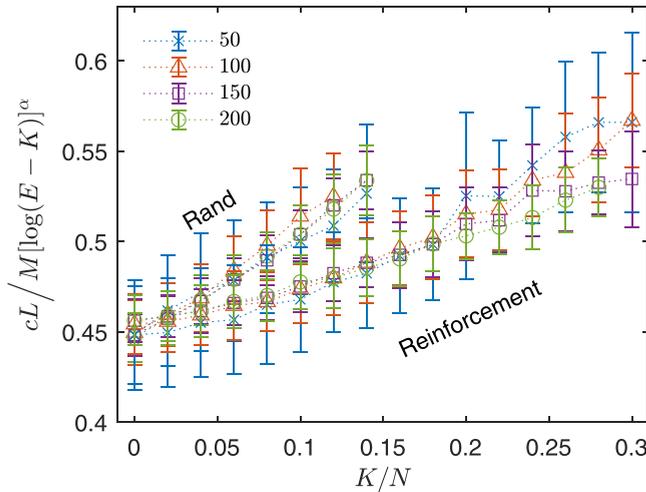


FIG. 6. Performance of reinforcement-based and random edge removal algorithms under node-disjoint routing for networks of size N . The power α and scale prefactor c for random edge removal are 1.4754 and 1, respectively, and for the reinforcement-based algorithm $\alpha = 1.3869$ and $c = 0.8763$, when the curves are adjusted to the same average starting value ($K = 0$) as random removal. The logarithm in the figure is of base e . From the current figure and Fig. 3(a), it is clear that our algorithm shows better performance than random edge removal. Each point is based on at least 15 samples.

ties. The corresponding probabilities are modified as

$$\begin{aligned} q_{i,j}^{a,\mu}(\sigma) &\propto \eta_{i,j}(|\sigma|) r_{i,j}^{a,\mu}(\sigma) p_{i \rightarrow j}^{a,\mu}(\sigma) p_{j \rightarrow i}^{a,\mu}(-\sigma), \\ q_{i \rightarrow j}^{a,\mu}(\sigma) &\propto \eta_{i,j}(|\sigma|) r_{i,j}^{a,\mu}(\sigma) p_{i \rightarrow j}^{a,\mu}(\sigma), \\ \tilde{r}_{i,j}^{a,\mu}(\sigma) &\propto \eta_{i,j}(|\sigma|) p_{i \rightarrow j}^{a,\mu}(\sigma) p_{j \rightarrow i}^{a,\mu}(-\sigma). \end{aligned} \quad (\text{B8})$$

The reinforcement algorithm used is based on BP iterations with reinforcement up to a certain number of times, followed by calculation of the removal probability $\{r_{i,j}\}$, and inference of the edge configuration \vec{c} accordingly, where $c_{i,j} = \arg \max_{c=0,1} r_{i,j}(c)$. If $|\vec{c}| = E - K$, use the routing algorithm [14] to check whether the residual network is able to accommodate the requests or not. If routes are allocated successfully, return the edges of state 0 in the configuration \vec{c} , otherwise repeat the BP iterations several times, and check again.

Results of numerical experiments for random regular networks of degree 3, different network sizes and number of removed edges are shown in Fig. 6.

APPENDIX C: EDGE REMOVAL IN EDGE-DISJOINT ROUTING

The equations of $r_{i,j}$ and $r_{i,j}^{a,\mu}$ for edge disjoint are similar to Eqs. (B2) and (B3), where auxiliary quantities $\tilde{r}_{i,j}^{a,\mu}$ should be modified due to the constraint on edge routing variables

$$\tilde{r}_{i,j}^{a,\mu}(\sigma) \propto t_{i,j}^{a,\mu}(|\sigma|) p_{i \rightarrow j}^{a,\mu}(\sigma) p_{j \rightarrow i}^{a,\mu}(-\sigma), \quad (\text{C1})$$

where $r_{i,j}^{a,\mu}$ has similar meaning as $r_i^{a,\mu}$ in the node-disjoint scenario.

Enforcing the edge-disjoint constraint, similar to Eq. (B4), edge-disjoint messages take the form

$$\begin{aligned} t_{i,j}^{a,\mu}(0) &\propto \prod_{v \neq \mu} \tilde{r}_{i,j}^{a,v}(0) + e^{-\beta w_{i,j}} \sum_{v \neq \mu} \tilde{r}_{i,j}^{a,v}(1) \prod_{\omega \neq \mu, v} \tilde{r}_{i,j}^{a,\omega}(0), \\ t_{i,j}^{a,\mu}(1) &\propto e^{-\beta w_{i,j}} \prod_{v \neq \mu} \tilde{r}_{i,j}^{a,v}(0), \end{aligned} \quad (\text{C2})$$

where an arbitrary cost $w_{i,j}$ of using edge (i, j) is included (this can be length, signal-to-noise ratio or any other edge cost). The probabilities of wavelength a on edge (i, j) for request μ without the influence from edge-disjoint constraint, similar to Eq. (B5), are obtained by

$$\begin{aligned} \tilde{r}_{i,j}^{a,\mu}(0) &\propto r_{i,j}^{a,\mu}(0) p_{i \rightarrow j}^{a,\mu}(0) p_{j \rightarrow i}^{a,\mu}(0), \\ \tilde{r}_{i,j}^{a,\mu}(1) &\propto \sum_{\sigma = \pm 1} r_{i,j}^{a,\mu}(\sigma) p_{i \rightarrow j}^{a,\mu}(\sigma) p_{j \rightarrow i}^{a,\mu}(-\sigma). \end{aligned} \quad (\text{C3})$$

1. Belief-propagation iteration and reinforcement

The BP equation set for multiwavelength edge-disjoint routing problem is given by

$$\begin{aligned} p_{i \rightarrow j}^{a,\mu}(0) &\propto \prod_{k \in \partial i \setminus j} q_{k \rightarrow i}^{a,\mu}(0) \\ &+ \sum_{m, n \in \partial i \setminus j} q_{m \rightarrow i}^{a,\mu}(1) q_{n \rightarrow i}^{a,\mu}(-1) \\ &\times \prod_{k \in \partial i \setminus j, m, n} q_{k \rightarrow i}^{a,\mu}(0), \\ p_{i \rightarrow j}^{a,\mu}(\pm 1) &\propto \sum_{k \in \partial i \setminus j} q_{k \rightarrow i}^{a,\mu}(\pm 1) \prod_{l \in \partial i \setminus j, k} q_{l \rightarrow i}^{a,\mu}(0); \\ q_{i \rightarrow j}^{a,\mu}(\sigma) &\propto r_{i,j}^{a,\mu}(\sigma) t_{i,j}^{a,\mu}(|\sigma|) p_{i \rightarrow j}^{a,\mu}(\sigma). \end{aligned} \quad (\text{C4})$$

The messages from auxiliary vertices for the wavelength selection are calculated by Eq. (B7).

We notice that quantities $r_{i,j}^{a,\mu}$ and $t_{i,j}^{a,\mu}$ are interlinked, to solve the equations, one can do the following steps:

0. initialize the quantities $\{q_{\rightarrow}^{\mu}\}$, $\{r_{i,j}^{a,\mu}\}$;
1. taking $\{q_{\rightarrow}^{\mu}\}$, calculate $\{p_{\rightarrow}^{\mu}\}$;
2. using $\{p_{\rightarrow}^{\mu}\}$ and $\{r_{i,j}^{a,\mu}\}$, calculate $\{t_{i,j}^{a,\mu}\}$;
3. calculate $\{\tilde{r}_{i,j}^{a,\mu}\}$, then λ and $\{r_{i,j}^{a,\mu}\}$;
4. do steps 2–3 several times, and compute $\{q_{\rightarrow}^{\mu}\}$ by taking $\{r_{i,j}^{a,\mu}, t_{i,j}^{a,\mu}, p_{\rightarrow}^{\mu}\}$.

Repeat steps 1–4 until convergence or a maximum number of steps to calculate the related probabilities and quantities of interest.

With the introduction of reinforcement factor $\eta_{i,j}$, the corresponding probabilities are modified as

$$\begin{aligned} q_{i,j}^{a,\mu}(\sigma) &\propto \eta_{i,j}(|\sigma|) r_{i,j}^{a,\mu}(\sigma) t_{i,j}^{a,\mu}(|\sigma|) p_{i \rightarrow j}^{a,\mu}(\sigma) p_{j \rightarrow i}^{a,\mu}(-\sigma), \\ q_{i \rightarrow j}^{a,\mu}(\sigma) &\propto \eta_{i,j}(|\sigma|) r_{i,j}^{a,\mu}(\sigma) t_{i,j}^{a,\mu}(|\sigma|) p_{i \rightarrow j}^{a,\mu}(\sigma), \\ \tilde{r}_{i,j}^{a,\mu}(\sigma) &\propto \eta_{i,j}(|\sigma|) t_{i,j}^{a,\mu}(|\sigma|) p_{i \rightarrow j}^{a,\mu}(\sigma) p_{j \rightarrow i}^{a,\mu}(-\sigma), \\ \tilde{r}_{i,j}^{a,\mu}(|\sigma|) &\propto \eta_{i,j}(|\sigma|) \sum_{\sigma = \pm |\sigma|} r_{i,j}^{a,\mu}(\sigma) p_{i \rightarrow j}^{a,\mu}(\sigma) p_{j \rightarrow i}^{a,\mu}(-\sigma). \end{aligned} \quad (\text{C5})$$

From our numerical experiments, as one increases the growth of the reinforcement factor η more rapidly using higher ε in Eq. (4) of the main text, the number of iterations decreases

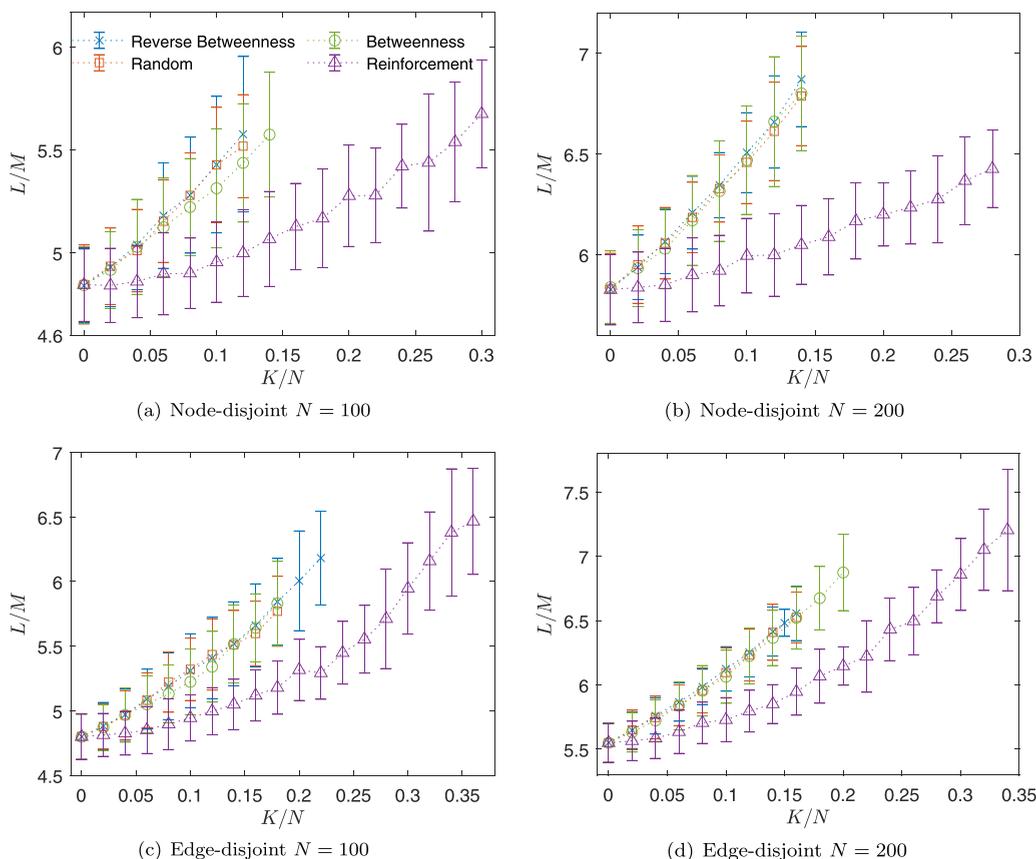


FIG. 7. Average path lengths of the various edge-removal methods under node- and edge-disjoint routing for $N = 100$ and 200 nodes random regular networks of degree 3. In the experiments, the number of wavelengths is $Q = 10$ and the values of corresponding number of communication requests M are chosen according to Table I. Each point is based on more than 15 samples.

at the increasing risk of failure to obtain good solutions. We therefore experimented with various values of ϵ .

2. Computational complexity

Clearly, the computational complexity of the algorithms we introduce impacts on their suitability to larger networks. We therefore summarize the computation effort in the different cases per iteration:

1. Simple routing scenario (path independent routing) - $O(M^2E)$.
2. Multiwavelength disjoint routing is approximately - $O((M + Q)M^2EQ)$.

where in all the cases E , M and Q are the number of edges, communication requests and wavelengths, respectively.

3. Betweenness centrality order for edge removal

Betweenness centrality is a commonly used measure to evaluate the importance of vertices and edges [32,39] on a given graph, defined as

$$B_{i,j} = \sum_{s \neq d} \frac{n_{i,j}^{sd}}{g^{sd}}, \tag{C6}$$

where g^{sd} is the number of (weighted) shortest paths from s to d , and $n_{i,j}^{sd}$ is the number of the shortest paths g^{sd} passing through edge (i, j) . Generally, the removal of edges with higher $B_{i,j}$ values would have a high impact on the average path lengths. In our case, it is reasonable to remove edges

with the lowest $B_{i,j}$ values to have minimal impact on the objective function, which we term removal using the reverse betweenness centrality order.

Interestingly, it seems from our numerical experiments that the order (betweenness centrality, reverse betweenness centrality and random) of edge removal does not make big difference on performance, as shown in Figs. 7 and 8. It is probably since the tested networks are random regular and edges are unweighted, which weaken the relevance of betweenness centrality. From the numerical simulation results, our algorithm exhibits the lowest average path length L/M [in Fig. 7(d)] and can remove far more edges (K_{\max}) than the other edge-removal protocols as shown in Fig. 8. This is not surprising since our algorithm does not rely merely on topological properties but takes the requested communication paths into account.

APPENDIX D: NETWORK GROWTH

We take edge-disjoint routing as an example, where the newly introduced edge set is S' of size $|S'| \geq K$, the latter being the number of edges added. As explained in the main text, the selection of the edge set S' from all available edges outside the original edge set of size E is carried out by examining the impact of individual additions on reducing the routing cost. The constraint on the number of edges added takes the form

$$\sum_{(i,j) \in S'} r_{i,j}(1) = K. \tag{D1}$$

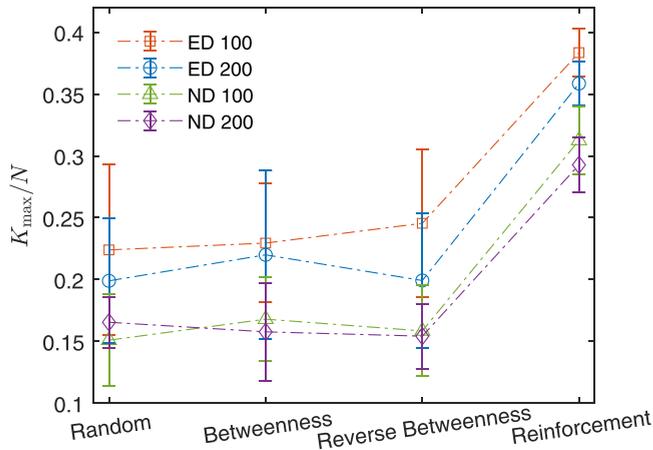


FIG. 8. Performance of the various edge removal methods for edge-disjoint (ED) and node-disjoint (ND) routing for different network sizes $N = 100$ and 200 . The parameters used are the same as in Fig. 7 and each point is based on 20 samples.

It means that the calculation of λ in each iteration depends only on the edges introduced in the edge set S' , and the routing variables $\sigma_{i,j}^{a,\mu}$ on the original edge set are not directly impacted by edge “removal” (addition) linked to variable $c_{i,j}$, equivalently, $c_{i,j} = 1 \forall (i, j) \notin S'$, or that the influence of variables $c_{i,j}$ on messages are all constant, such that $t_{i,j}^{a,\mu}(0) = t_{i,j}^{a,\mu}(1) = 1/2, \forall (i, j) \notin S'$.

Expansion algorithms

Three algorithms are used for selecting K edges from the set S' :

- (1) *random*. randomly select K edges out of S' ;
- (2) *greedy*. select a previously unselected edge in S' , which reduces the average path length the most; repeat K times;
- (3) *decimation*. follow the steps in the decimation algorithm of the average path length section, but in the decimation step, fix the edge state $c_{i,j} = 1$ with the largest $r_{i,j}(1) \in S'$ value.

- [1] R. Albert, H. Jeong, and A.-L. Barabási, Error and attack tolerance of complex networks, *Nature (London)* **406**, 378 (2000).
- [2] R. Albert and A.-L. Barabási, Statistical mechanics of complex networks, *Rev. Mod. Phys.* **74**, 47 (2002).
- [3] M. E. J. Newman, The structure and function of complex networks, *SIAM Rev.* **45**, 167 (2003).
- [4] S. N. Dorogovtsev, A. V. Goltsev, and J. F. F. Mendes, Critical phenomena in complex networks, *Rev. Mod. Phys.* **80**, 1275 (2008).
- [5] J.-P. Onnela, J. Saramäki, J. Hyvönen, G. Szabó, D. Lazer, K. Kaski, J. Kertész, and A.-L. Barabási, Structure and tie strengths in mobile communication networks, *Proc. Natl. Acad. Sci.* **104**, 7332 (2007).
- [6] P. Yuan and A. Xu, The influence of physical network topologies on wavelength requirements in optical networks, *J. Lightwave Technol.* **28**, 1338 (2010).
- [7] R. Matzner, D. Semrau, R. Luo, G. Zervas, and P. Bayvel, Making intelligent topology design choices: understanding structural and physical property performance implications in optical networks [invited], *J. Opt. Commun. Netw.* **13**, D53 (2021).
- [8] R. Luo, R. Matzner, G. Zervas, and P. Bayvel, Towards a traffic-optimal large-scale optical network topology design, in *2022 International Conference on Optical Network Design and Modeling (ONDM)* (IEEE, Piscataway, NJ, 2022), pp. 1–3.
- [9] B. Li, D. Saad, and C. H. Yeung, Bilevel optimization in flow networks: A message-passing approach, *Phys. Rev. E* **106**, L042301 (2022).
- [10] D. S. Johnson, J. K. Lenstra, and A. H. G. R. Kan, The complexity of the network design problem, *Networks* **8**, 279 (1978).
- [11] R. M. Karp, Reducibility among combinatorial problems, in *Complexity of Computer Computations*, edited by R. E. Miller, J. W. Thatcher, and J. D. Bohlinger (Springer US, Boston, MA, 1972), pp. 85–103.
- [12] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, 1st ed. (W. H. Freeman, New York, 1979).
- [13] B. Korte and J. Vygen, *Multicommodity flows and edge-disjoint paths*, in *Combinatorial Optimization: Theory and Algorithms* (Springer, Berlin, Heidelberg, 2012), pp. 489–520.
- [14] Y.-Z. Xu, H. F. Po, C. H. Yeung, and D. Saad, Scalable node-disjoint and edge-disjoint multiwavelength routing, *Phys. Rev. E* **105**, 044316 (2022).
- [15] E. K. Çetinkaya, M. J. Alenazi, Y. Cheng, A. M. Peck, and J. P. Sterbenz, A comparative analysis of geometric graph models for modelling backbone networks, *Opt. Switch. Netw.* **14**, 95 (2014).
- [16] J. Velinska, M. Mirchev, and I. Mishkovski, Optical networks’ topologies: costs, routing and wavelength assignment, *ICT Innovations* 1 (2017).
- [17] Y. Kabashima and D. Saad, Belief propagation vs. TAP for decoding corrupted messages, *Europhys. Lett.* **44**, 668 (1998).
- [18] M. Mézard and A. Montanari, *Information, Physics, and Computation* (Oxford University Press, Oxford, UK, 2009).
- [19] C. De Bacco, S. Franz, D. Saad, and C. H. Yeung, Shortest node-disjoint paths on random graphs, *J. Stat. Mech.: Theory Exp.* (2014) P07009.
- [20] F. Altarelli, A. Braunstein, L. Dall’Asta, C. De Bacco, and S. Franz, The edge-disjoint path problem on random graphs by message-passing, *PLoS ONE* **10**, e0145222 (2015).
- [21] 60-node network derived from the coronet conus topology, the data was originally downloaded from the web site <http://monarchna.com/60-Node-CONUS-Topology.xls> but is also available on [31].
- [22] P. Šulc and L. Zdeborová, Belief propagation for graph partitioning, *J. Phys. A: Math. Theor.* **43**, 285003 (2010).
- [23] Y.-Z. Xu and H.-J. Zhou, Optimal segmentation of directed graph and the minimum number of feedback arcs, *J. Stat. Phys.* **169**, 187 (2017).
- [24] C. H. Yeung and D. Saad, Competition for Shortest Paths on Sparse Graphs, *Phys. Rev. Lett.* **108**, 208701 (2012).
- [25] C. H. Yeung, D. Saad, and K. M. Wong, From the physics of interacting polymers to optimizing routes on the london underground, *Proc. Natl. Acad. Sci. USA* **110**, 13717 (2013).

- [26] H. A. Bethe, Statistical theory of superlattices, *Proc. R. Soc. London A* **150**, 552 (1935).
- [27] M. Mézard and G. Parisi, The Bethe lattice spin glass revisited, *Eur. Phys. J. B* **20**, 217 (2001).
- [28] H.-J. Zhou, *Spin Glass and Message Passing* (Science Press, Beijing, China, 2015).
- [29] M. E. J. Newman, Message passing methods on complex networks, *Proc. R. Soc. A* **479**, 20220774 (2023).
- [30] A. Braunstein and R. Zecchina, Learning by Message Passing in Networks of Discrete Synapses, *Phys. Rev. Lett.* **96**, 030201 (2006).
- [31] Data available on <https://github.com/XuYZh/Network-Pruning-and-Growth—Probabilistic-Optimization>.
- [32] L. Freeman, A set of measures of centrality based on betweenness, *Sociometry* **40**, 35 (1977).
- [33] H. Yang and M. G. H. Bell, Models and algorithms for road network design: a review and some new developments, *Transp. Rev.* **18**, 257 (1998).
- [34] M. Cutler and Y. Shiloach, Permutation layout, *Networks* **8**, 253 (1978).
- [35] A. Aggarwal, J. Kleinberg, and D. P. Williamson, Node-disjoint paths on the mesh and a new trade-off in vlsi layout, *SIAM J. Comput.* **29**, 1321 (2000).
- [36] L. Wang, M. Klein, S. Yirga, and P. Kundur, Dynamic reduction of large power systems for stability studies, *IEEE Trans. Power Syst.* **12**, 889 (1997).
- [37] J. Yang, G. Cheng, and Z. Xu, *Dynamic reduction of large power system in pss/e*, in *2005 IEEE/PES Transmission and Distribution Conference & Exposition: Asia and Pacific* (IEEE, Piscataway, NJ, 2005), pp. 1–4.
- [38] P. Erdrich, R. Steuer, and S. Klamt, An algorithm for the reduction of genome-scale metabolic network models to meaningful core models, *BMC Syst. Biol.* **9**, 48 (2015).
- [39] M. Newman, *Networks: An Introduction*, 1st ed. (Oxford University Press, Oxford, UK, 2010).