**Abstract**

A new solution framework for the task of network dismantling is recently developed, based on a two-scale bipartite factor-graph representation of the original graph where local structures are abstracted as factor nodes. This technique leads to advancement of extant dismantling algorithms, among which the belief-propagation decimation (BPD) algorithm has an efficient counterpart (factor BPD, i.e., FBPD) on the factor graph, building upon a mean-field spin-glass theory developed for the underlying long-loop feedback vertex set (FVS) problem. In this paper, I (1) demonstrate the advantage as well as disadvantage of the new factor-graph approach, and investigate the varying choice of factors, (2) show that the method can be supported by an alternative microscopic picture, and the two distinct spin-glass theories derive equivalent outcomes, whose analytical results serve as lower bounds for the FVS size on random regular factor graphs, besides (3) an extra mathematical lower bound from the result on random regular (original) graphs. Performances of graph/factor-graph algorithms are compared on various real networks. It shows empirically and analytically that the factor-graph approach does not interfere with what we could achieve without applying this technique; the new approach does a good job where traditional algorithms may perform poorly.

Many effective algorithms have been developed in the last decade for the task of network dismantling, namely, with node removal as minimal as possible, to break the entire network into small subgraphs whose sizes fall below a certain threshold. State-of-the-art algorithms include EGP [1], CI [2], Min-Sum [3], belief-propagation decimation (BPD) [4], CoreHD [5], EI [6] etc (see a comparative analysis in [7]). Recent studies explore new ideas for this problem, such as spectral graph bisection for ‘generalized network dismantling’ [8], or dismantling based on machine learning techniques [9, 10]. A key idea for network dismantling [3–5] is to start with finding an efficient feedback vertex set (FVS) of the graph whose removal gives rise to a decycled remainder graph, and then conduct tree-breaking and re-insertion procedures on the resulting subgraphs under the size threshold. By this means, the task of network dismantling is coupled with the task of network decycling, and one expects that an efficient network decycling algorithm which determines near-optimal FVS would further lead to an effective network dismantling algorithm. The spin-glass theory determines the optimal FVS on trees and is quasi-optimal on graphs with long-loops [11], which leads to the BPD algorithm for dismantling [4]. However, BPD is not optimal for the FVS identification on structures that deviate much from trees (e.g., having many short loops), and it is suggested that room is clearly left for further improvement upon BPD results, especially when dealing with real networks which often exhibit non-trivial clustering [12–14].

Recently, a novel dismantling solution framework based on factor-graph representation [15] is proposed, which enhances the power of BPD by overlooking dense local structures/factors [16, 17] of the graph while focusing on the more sparse long loops. Those factors are represented with a new type of nodes (i.e., factor nodes), and the original graph is transformed into a topologically simplified bipartite factor
We consider cliques (among which triangles/three-cliques are the most common type) as graph factors and represent them with factor nodes, which transform the original graph to the bipartite factor graph (Figure 1). By this means, local factors disappear and the graph is simplified to a tree if it does not contain any four-cliques and corresponding network dismantling results are expected (yet not guaranteed, see below) to improve through efficient message-passing. Suppose the original graph $G = (N, E)$ and the factor graph $G' = (N', E')$. We denote the set of real nodes and factor nodes with $R = \{i, j, k, \ldots \}$ and $I = \{a, b, c, \ldots \}$, i.e., $|R| = N$ and $|R| + |I| = N'$ (no real node is deleted in the factor graph). Note that during graph transformation, all original edges $E$ are broken and any edge in the factor graph $G'$ is always between a real node and a factor node, i.e., there is only one type of edges in the factor graph despite two types of nodes.

Factor nodes in figure 1 are shown with squares, among which a simple edge is a two-clique, a triangle is a three-clique (blue), and a tetrahedron is a four-clique (yellow).

Table 1. Factor graphs of real networks and the FVS size ($m$). Superscripts $B$, $FB$, $G$, $FC$ are short for BPD, FBPD, CoreHD, FCoreHD; subscripts stand for the values of $\gamma$. Best (smallest) $m$ of each row is shown in bold.

| Network | $|N|$ | $|N'_b|$ | $|N'_f|$ | $|E|$ | $|E'_b|$ | $|E'_f|$ | $m^B$ | $m^{B'}_{15}$ | $m^{B'}_{20}$ | $m^{B'}_{25}$ | $m^{B'}_{30}$ | $m^{C}$ | $m^{C'}_{15}$ | $m^{C'}_{20}$ | $m^{C'}_{25}$ | $m^{C'}_{30}$ |
|---------|------|-------|-------|------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| Email1  | 986  | 8566  | 7416  | 7261 | 16 064 | 19 402 | 17 012 | 16 518 | 449   | 448   | 448   | 442   | 440   | 439   | 436   | 427   | 418   |
| Yeast   | 2284 | 7722  | 7657  | 7661 | 6646   | 11 480 | 11 338 | 11 330 | 362   | 352   | 349   | 349   | 349   | 346   | 343   | 340   | 339   | 335   |
| FB-a    | 4039 | 40 437 | 30 770 | 28 677 | 88 234 | 98 714 | 79 273 | 73 509 | 2445  | 2425  | 2386  | 2364  | 2444  | 2419  | 2386  |       |       |       |
| Grid    | 4941 | 10 803 | 10 735 | 10 729 | 6 594   | 12 090 | 11 972 | 11 961 | 520   | 447   | 429   | 422   | 421   | 416   | 412   | 412   | 409   | 407   |
| IntNet  | 674  | 17 634 | 17 604 | 17 608 | 12 572  | 23 026 | 22 960 | 22 957 | 215   | 195   | 192   | 192   | 195   | 192   | 192   | 195   | 195   | 193   |
| LastFM  | 7624 | 25 952 | 25 054 | 24 940 | 12 806  | 23 026 | 22 960 | 22 957 | 215   | 195   | 192   | 192   | 195   | 192   | 192   | 195   | 195   | 193   |
| Tweet   | 9498 | 107 166 | 105 222 | 105 096 | 153 138 | 223 071 | 218 071 | 217 395 | 3067  | 3048  | 3040  | 3033  | 3057  | 3065  | 3065  |       |       |       |
| FB-b    | 13 866 | 71 153 | 69 295 | 69 053 | 86 811  | 129 536 | 125 214 | 124 430 | 4218  | 4123  | 4095  | 4070  | 4029  | 4106  | 4128  | 4128  |       |       |
| Email2  | 36 692 | 138 985 | 126 826 | 126 826 | 183 831 | 245 355 | 221 045 | 221 045 | 7860  | 7134  | 5674  | 5674  | 7846  | 7104  | 5668  | 5668  |       |       |
| Deezel1 | 47 538 | 215 761 | 212 092 | 201 686 | 222 887 | 363 778 | 356 686 | 355 602 | 17 343 | 17 021 | 16 942 | 16 930 | 17 644 | 17 340 | 17 271 | 17 265 |       |       |
| Deezel2 | 54 573 | 376 263 | 362 926 | 351 790 | 498 202 | 731 636 | 703 980 | 700 359 | 24 609 | 24 377 | 24 332 | 24 311 | 24 958 | 24 739 | 24 696 | 24 702 |       |       |
| BK      | 58 228 | 208 928 | 203 215 | 202 580 | 214 078 | 333 089 | 321 338 | 319 346 | 847 776 | 7621  | 7570  | 8497  | 7811  | 7658  | 7599  |       |       |       |

Figure 1. Factor graph transformation (adapted from figure 1 in [15]). A simple edge is a two-clique (red); a triangle is a three-clique (blue), and a tetrahedron is a four-clique (yellow).
Figure 2. Improved dismantling performance on a small sample graph (see text for explanation).

Figure 3. Downgraded dismantling performance with the factor-graph approach. For task |LCC| ≤ 5, when conducting dismantling on the original graph, the optimal solution is the removal of 3 nodes (shown in light shade); on the corresponding factor graph, 5 nodes need to be removed (not shown). Not capturing incomplete cliques (such as the structure on nodes A to D) under the current scheme undermines the power of the new approach.

three single nodes, which could not formulate a triangle due to the missing edge. Each of the three nodes connects to the 5 nodes in the clique, and there is 2 edges connecting them three. Therefore, there are \(1 + 3 \times 5 + 2 = 18\) factor nodes and \(5 + 2 \times (3 \times 5 + 2) = 39\) edges in total. In this case, elevating the clique order from \(\gamma = 4\) to \(\gamma = 5\) leads to more nodes and edges in the factor graph. This issue is resolved if the clique order is further elevated, e.g., \(\gamma \geq 7\) yields the optimal factor-graph representation of this example. Such phenomena of insufficient clique order cutoffs, however, do not appear at most graph instances that we tested in this study, but for very few exceptions (see Table 1).

Improvement of (decycling-based) dismantling performance on the factor graph versus on the original graph can be demonstrated on the sample graph in Figure 2. The seven-node graph is connected through 2 triangles and 2 single edges, whose corresponding factor graph is a tree consisting of 11 nodes. When conducting dismantling on the original graph under the condition that the resulting largest connected component (LCC) has |LCC| ≤ 3, two nodes (C and E) need to be removed at the first step to render the graph decycled. Neither of the two nodes can be inserted back without violating the LCC size constraint; thus the final removal set has length 2. Dismantling the (already acyclic) factor graph calls directly for tree-breaking, where only one node (D) needs to be put into the removal set before the LCC size constraint is met. This small example reveals the potential of the factor-graph approach in dismantling real networks, where local clusters consisting of triangles and higher-order cliques often prevail.

Nevertheless, the new method does not always improve dismantling performance and may occasionally lead to downgraded performance (Figure 3). A critical failure lies in the inability of capturing incomplete cliques (e.g., nodes A to D in Figure 3) under the current scheme where only cliques are identified as factors. An incomplete clique is not able to be successfully abstracted as a factor, and has to be represented with a lower clique plus additional edges. In future extensions, it is plausible to explore the incorporation of incomplete cliques as new factors and evaluate the method’s updated performance. Initial tests (see later) suggest that capturing more factor types does render the factor graph more simplified, yet the eventual dismantling performance can be downgraded. This is because the set of re-insertable nodes that are removed at the decycling step drastically shrinks as now individual nodes are associated with more complex factors; a possible re-insertion may often incur putting back a large number of nodes and is thus inhibited.
2. Lower bound of FVS size on random regular factor graphs

It is expected that on random regular factor graphs, the new approach will lead to smaller FVS size compared with that obtained under the normal graph approach. We derive lower bounds of FVS size on random regular factor graphs. Start with random regular triangle graphs [18, 19]. On these graphs each node has exactly $K$ triangles, and each triangle in the graph is formed with a tuple of three nodes selected at random; the graph contains no other structure, not excluding simple edges. Thus in the factor graph, each factor node has exactly 3 neighbors and each real node has $K$ neighbors. Suppose there is $N$ real nodes, then the number of factor nodes added to the graph is $KN/3$. For random regular (non-triangle) graphs of order $K$, the lower bound of FVS size (as a fraction of node has exactly 3 neighbors and each real node has $K$ neighbors. Suppose there is $N$ real nodes, then the number of factor nodes added to the graph is $KN/3$. For random regular (non-triangle) graphs of order $K$, the system is close to regular non-triangle graphs of order $2K$ (i.e., each node has $K$ triangles and thus $2K$ neighbors). However, after $b(2K)$ nodes have been removed, the remaining graph is a tree structure in terms of local triangles, thanks to the breakdown of long loops. Since local triangles still exist, the lower bound of FVS size $b^{\text{tri}}(K)$ is slightly greater than $b(2K)$, with additional nodes to be removed in order to break local structures:

$$b^{\text{tri}}(K) = b(2K) + \Delta b.$$  \hspace{1cm} (1)

For the lower bound, we want to find the minimum number of $\Delta b$ that once these nodes are removed, the remaining structure is completely acyclic. After the removal of $bN$ nodes, there are $(1-b)N$ nodes left, organized as a tree when neglecting local triangles. The minimum number of nodes required to break all local triangles on such a quasi-tree, is obtained when the structure is most densely organized, i.e., starting from a central node, every node has the same number of triangles organized layer-wise, similar to a Bethe lattice, with the mean-field triangle degree given by $K' = (1-b)K$. Then, when all inner-layer nodes are removed, the structure becomes completely acyclic, obtaining the most efficient removal. Counting the nodes at each layer (up to a certain layer $w$), we have $1 + 2K' + 2K' \cdot 2(K' - 1) + 2K' \cdot 2(K' - 1) \cdot 2(K' - 1) + \cdots = 1 + 2K' \sum_{i=0}^{w-1} (2(K' - 1))^i = (1-b)N$. Therefore

$$\Delta b = \frac{1}{N} \sum_{i=0}^{w-1} [2(K' - 1)]^i = \frac{1-b}{4K'(K' - 1)} - \frac{1 + 2K'}{4K'(K' - 1)N}.$$  \hspace{1cm} (2)

In the limit of $N \to \infty$, we arrive at a lower bound of FVS size on random regular triangle graphs of order $K$:

$$b^{\text{tri}}(K) = b(2K) + \frac{1-b(2K)}{4K'(K' - 1)}$$

$$= b(2K) + \frac{1}{4K[1-b(2K)]}.$$  \hspace{1cm} (3)

where $b(2K)$ is given by [20]. Following similar derivations, for random regular tetrahedron graphs, one arrives at the lower bound:

$$b^{\text{qua}}(K) = b(3K) + \frac{1-b(3K)}{9K(K' - 1)}$$

$$= b(3K) + \frac{1}{9K[1-b(3K)]}.$$  \hspace{1cm} (4)

which could further extend to random regular factor graphs with higher order of cliques being the factor.

3. Belief-propagation on factor graphs: theory A

A spin-glass theory is constructed for the factor-graph model based on a microscopic thermodynamic picture [15]. Here we explain the theory in detail. Suppose each real node $i$ has the direction state $s_i$ which could be zero, i.e., $s_i = 0$, indicating that $i$ being deleted from the graph, or it could point to one of the neighboring factor nodes $a \in \partial i$ and then $s_i = a$. A microscopic configuration of the entire system of $N$ nodes (considering only the states of real nodes, not factor nodes) is denoted as $\mathbf{s} \equiv (s_1, s_2, \ldots, s_N)$. Each factor node $a$ then serves as a constraint on the configuration $\mathbf{s}$ of the system:

$$\chi_a = \prod_{i \in \partial a} (\delta_{s_i}^a + \delta_{s_i}^0) + \sum_{i \in \partial a} (1 - \delta_{s_i}^a - \delta_{s_i}^0) \prod_{j \in \partial a \setminus i} (\delta_{s_j}^a + \delta_{s_j}^0),$$  \hspace{1cm} (5)
where $\delta$ is the Kronecker delta: $\delta_{x}^{y} = 1$ iff $x = y$, otherwise $\delta_{x}^{y} = 0$. Notice that $\chi_{a} = 1$ iff (i) every neighboring real node $i$ of node $a$ is in state $s_i = a$ or state $s_i = 0$, or (ii) exactly one neighboring node $i$ is not in state $s_i = a$ or $s_i = 0$ while all other neighboring nodes $j \in \partial a$ are all in state $s_j = a$ or $s_j = 0$. Now the partition function of the system is defined as

$$Z(\beta) = \sum_{a}^{N} \prod_{i=1}^{Z} \left[ e^{-\beta \delta_{i}^{a}} + (1 - \delta_{i}^{a}) \right] \prod_{v}^{M} \chi_{av},$$

(6)

where $\beta$ is the inverse temperature controlling the mean number of zero-state nodes in the system. Therefore, any valid configuration $\mathbf{z}$ of the system satisfies all constraints $\chi = 1$ induced by all existing factor nodes (noting the multiplication term at the end of equation (6)). When any constraint is not satisfied and we have $\chi = 0$ somewhere, it means that a certain factor node is not being properly considered, then the corresponding system configuration is not valid and will not be counted in $Z$.

Essentially, due to the constraints $\chi$ from factor nodes, every configuration $\mathbf{z}$ having positive contribution to $Z(\beta)$ can be represented as a set made of trees (each of which is a factor subgraph with no loop) and cycle-trees (each of which is a factor subgraph with exactly one loop); as long as a configuration $\mathbf{z}$ is not conflicting a tree or quasi-tree, it is valid (for example, a valid (local) configuration for nodes $F$, $G$, $K$ in figure 1 can be $s_{F/G/K} = 4/4/4$, or $s_{F/G/K} = 4/4/5$, $s_{F/G/K} = 2/4/4$, $s_{F/G/K} = 4/8/4$... As long as at least two of $F$, $G$, $K$ are pointing to the factor node 4, the configuration is valid).

We write the cavity probabilities of the state $s_i$ of each real node $i$ and then formulate the self-consistent BP equations under this spin-glass theory. Denote by $p_{a \rightarrow i}^{a}$ the cavity probability of state $s_i$ of node $i$ if $i$ is constrained only by factor node $a$ but not by all the other connected factor nodes $b$, and $q_{b \rightarrow i}^{a}$ the cavity probability of state $s_i$ of node $i$ if $i$ is constrained by all the connected factor nodes $b \in \partial i$ except for the factor node $a$. Self-consistent BP equations for $p_{a \rightarrow i}^{a}$ are

$$p_{a \rightarrow i}^{0} = \frac{1}{2z_{a ightarrow i}} \prod_{j \in \partial a \setminus i} \left( q_{j \rightarrow i}^{0} + q_{j \rightarrow i}^{a} \right) + \sum_{j \in \partial a \setminus i} \left( 1 - q_{j \rightarrow i}^{0} - q_{j \rightarrow i}^{a} \right) \prod_{k \in \partial a \setminus i} \left( q_{k \rightarrow i}^{0} + q_{k \rightarrow i}^{a} \right),$$

(7)

$$p_{a \rightarrow i}^{a} = \frac{1}{2z_{a ightarrow i}} \prod_{j \in \partial a \setminus i} \left( q_{j \rightarrow i}^{a} + q_{j \rightarrow i}^{0} \right) + \sum_{j \in \partial a \setminus i} \left( 1 - q_{j \rightarrow i}^{a} - q_{j \rightarrow i}^{0} \right) \prod_{k \in \partial a \setminus i} \left( q_{k \rightarrow i}^{a} + q_{k \rightarrow i}^{0} \right),$$

$$p_{a \rightarrow i}^{b} = \frac{1}{2z_{b ightarrow i}} \prod_{j \in \partial b \setminus i} \left( q_{j \rightarrow i}^{a} + q_{j \rightarrow i}^{b} \right), \quad (\text{for } b \in \partial i \setminus a).$$

The normalization constant $z_{a ightarrow i}$ ensures that $p_{a \rightarrow i}^{a} + p_{a \rightarrow i}^{0} + \sum_{b \in \partial i \setminus a} p_{a \rightarrow i}^{b} = 1$. Notice that $p_{a \rightarrow i}^{0} = p_{a \rightarrow i}^{j}$ for any $b \in \partial i \setminus a$ hence its value is $p_{a \rightarrow i}^{0} = \frac{1}{2z_{a \rightarrow i}}$. Thus cavity probabilities $p_{a \rightarrow i}^{a}$ can essentially be expressed with a single parameter (e.g., $p_{a \rightarrow i}^{0}$). Self-consistent BP equations for $q_{j \rightarrow i}^{a}$ are

$$q_{j \rightarrow i}^{0} = \frac{1}{z_{j \rightarrow i}} e^{-\beta} \prod_{b \in \partial j \setminus i} p_{b \rightarrow i}^{0},$$

$$q_{j \rightarrow i}^{a} = \frac{1}{z_{j \rightarrow i}} \prod_{b \in \partial j \setminus i} p_{b \rightarrow i}^{a},$$

(8)

$$q_{j \rightarrow i}^{b} = \frac{1}{z_{j \rightarrow i}} \prod_{c \in \partial j \setminus i,b} p_{c \rightarrow i}^{b}, \quad (\text{for } b \in \partial i \setminus a).$$

where $z_{i \rightarrow a}$ ensures $q_{i \rightarrow a}^{0} + q_{i \rightarrow a}^{a} + \sum_{b \in \partial i \setminus a} q_{i \rightarrow a}^{b} = 1$. The marginal probabilities $q_{i}^{a}$ of state $s_i$ of the real node $i$ are

$$q_{i}^{0} = \frac{1}{z_{i}} e^{-\beta} \prod_{a \in \partial i} p_{a \rightarrow i}^{0},$$

$$q_{i}^{a} = \frac{1}{z_{i}} p_{a \rightarrow i}^{0} \prod_{b \in \partial i \setminus a} p_{b \rightarrow i}^{a}, \quad (\text{for } a \in \partial i).$$

(9)

with $z_{i}$ such that $q_{i}^{0} + \sum_{a \in \partial i} q_{i}^{a} = 1$. Compared to the two types of free entropy contribution under the original BP scheme, under the factor-graph scheme there are three types: contribution from real nodes
\( \langle \phi_{\in R} \rangle \), contribution from factor nodes \( \langle \phi_{a \in I} \rangle \), and contribution from edges \( \langle \phi_{u} \rangle \) (noting the uniform edge type).

\[
\langle \phi_{\in R} \rangle = -\frac{1}{\beta} \ln \left[ e^{-\beta} \prod_{a \in \partial i} p_{a \rightarrow i}^0 + \sum_{a \in \partial i} p_{a \rightarrow i}^0 \prod_{b \in \partial \partial i} p_{b \rightarrow i}^0 \right],
\]

\[
\langle \phi_{a \in I} \rangle = -\frac{1}{\beta} \ln \left[ \prod_{i \in a} \left( q_{i \rightarrow a}^0 + q_{i \rightarrow a}^0 \right) + \sum_{i \in a} \left( 1 - q_{i \rightarrow a}^0 - q_{i \rightarrow a}^0 \right) \prod_{j \in \partial i} \left( q_{j \rightarrow a}^0 + q_{j \rightarrow a}^0 \right) \right],
\]

\[
\langle \phi_{u} \rangle = -\frac{1}{\beta} \ln \left[ \prod_{i \in a \in \partial i} \left( q_{i \rightarrow a}^0 + q_{i \rightarrow a}^0 \right) \prod_{b \in \partial \partial i} \left( q_{b \rightarrow a}^0 \right) \right].
\]

The total free entropy \( \Phi \) at a certain temperature is summing over the three terms:

\[
\Phi(\beta) = \sum_{i \in R} \langle \phi_{\in R} \rangle + \sum_{a \in I} \langle \phi_{a \in I} \rangle - \sum_{(a \in \partial i) \in \partial i} \langle \phi_{u} \rangle,
\]

and the density \( f = \Phi(\beta)/N \). The mean fraction of feedback vertices \( \rho = \frac{1}{N} \sum_{i=1}^{N} q_{i \rightarrow a}^0 \) and

\[
q_{i \rightarrow a}^0 = \frac{e^{-\beta} \prod_{a \in \partial i} p_{a \rightarrow i}^0 + \sum_{a \in \partial i} p_{a \rightarrow i}^0 \prod_{b \in \partial \partial i} p_{b \rightarrow i}^0}{e^{-\beta} \prod_{a \in \partial i} p_{a \rightarrow i}^0 + \sum_{a \in \partial i} p_{a \rightarrow i}^0 \prod_{b \in \partial \partial i} p_{b \rightarrow i}^0}.
\]

At a given value of \( \beta \) we obtain \( \rho \) and the overall entropy density of the system \( s = \beta(f - \rho) \). The minimum fraction \( \rho_0 \) of feedback vertices is thus determined when \( s(\rho_0) = 0 \).

The above equations are simplified when we only consider cavity messages \( q_{i \rightarrow a}^0 \), in which case the BP equations become

\[
q_{i \rightarrow a}^0 \propto e^{-\beta} \prod_{b \in \partial \partial i} \left[ \prod_{j \in \partial i} \left( d_{i \rightarrow j}^0 + d_{i \rightarrow j}^0 \right) + \sum_{j \in \partial i} \left( 1 - d_{i \rightarrow j}^0 - d_{i \rightarrow j}^0 \right) \prod_{k \in \partial \partial j} \left( d_{k \rightarrow j}^0 + d_{k \rightarrow j}^0 \right) \right],
\]

\[
q_{i \rightarrow a}^0 \propto \prod_{b \in \partial \partial i} \left[ \prod_{j \in \partial i} \left( d_{i \rightarrow j}^0 + d_{i \rightarrow j}^0 \right) \right],
\]

\[
q_{i \rightarrow a}^0 \propto \prod_{b \in \partial \partial i} \left[ \prod_{j \in \partial i} \left( d_{i \rightarrow j}^0 + d_{i \rightarrow j}^0 \right) \right],
\]

and further

\[
q_{i \rightarrow a}^0 \propto e^{-\beta} \prod_{b \in \partial \partial i} \left[ 1 + \sum_{j \in \partial \partial i} \left( \frac{1}{d_{i \rightarrow j}^0 + d_{i \rightarrow j}^0} - 1 \right) \right],
\]

\[
q_{i \rightarrow a}^0 \propto 1,
\]

with normalization \( q_{i \rightarrow a}^0 + q_{i \rightarrow a}^0 + \sum_{b \in \partial \partial i} q_{b \rightarrow a}^0 = 1 \). Free entropy \( \Phi(\beta) \) is now expressed as

\[
\Phi(\beta) = \sum_{i \in R} \langle \phi_{\in R} \rangle + \sum_{a \in I} \langle \phi_{a \in I} \rangle - \sum_{(a \in \partial i) \in \partial i} \langle \phi_{u} \rangle,
\]

where \( \langle \phi_{a \in I} \rangle \) of a factor node \( a \) is as in \( (10) \); \( \langle \phi_{a \in I} \rangle \) is the combined free energy contribution of real node \( i \) and all its neighboring factor nodes \( a \in \partial i \), which is given by

\[
\langle \phi_{a \in I} \rangle = -\frac{1}{\beta} \ln (z_{i \rightarrow a}^0),
\]
with $z_{i+\partial i}$ expressed as

$$
z_{i+\partial i} = e^{-\beta} \prod_{a \in \partial i} \left[ \prod_{j \in \partial a \setminus i} (q_{f_{j\rightarrow a}}^{0} + q_{f_{j\rightarrow a}}^{0}) + \sum_{j \in \partial a \setminus i} (1 - q_{f_{j\rightarrow a}}^{0} - q_{f_{j\rightarrow a}}^{0}) \prod_{k \in \partial a \setminus i} (q_{f_{k\rightarrow a}}^{0} + q_{f_{k\rightarrow a}}^{0}) \right] + \sum_{a \in \partial i} \left[ \prod_{j \in \partial a \setminus i} (q_{f_{j\rightarrow a}}^{0} + q_{f_{j\rightarrow a}}^{0}) + \sum_{j \in \partial a \setminus i} (1 - q_{f_{j\rightarrow a}}^{0} - q_{f_{j\rightarrow a}}^{0}) \prod_{k \in \partial a \setminus i} (q_{f_{k\rightarrow a}}^{0} + q_{f_{k\rightarrow a}}^{0}) \right] \times \prod_{b \in \partial(i \cup a)} \left[ \prod_{k \in \partial b \setminus j} (q_{f_{k\rightarrow b}}^{0} + q_{f_{k\rightarrow b}}^{0}) \right].
$$

Now the marginal probability of node $i$ being empty is

$$
q_{i}^{0} = \frac{1}{z_{i+\partial i}} e^{-\beta} \prod_{a \in \partial i} \left[ \prod_{j \in \partial a \setminus i} (q_{f_{j\rightarrow a}}^{0} + q_{f_{j\rightarrow a}}^{0}) + \sum_{j \in \partial a \setminus i} (1 - q_{f_{j\rightarrow a}}^{0} - q_{f_{j\rightarrow a}}^{0}) \prod_{k \in \partial a \setminus i} (q_{f_{k\rightarrow a}}^{0} + q_{f_{k\rightarrow a}}^{0}) \right] = \frac{e^{-\beta} \prod_{a \in \partial i} \left[ 1 + \sum_{j \in \partial a \setminus i} \left( \frac{1}{q_{f_{j\rightarrow a}}^{0} + q_{f_{j\rightarrow a}}^{0}} - 1 \right) \right]}{e^{-\beta} \prod_{a \in \partial i} \left[ 1 + \sum_{j \in \partial a \setminus i} \left( \frac{1}{q_{f_{j\rightarrow a}}^{0} + q_{f_{j\rightarrow a}}^{0}} - 1 \right) \right] + \sum_{a \in \partial i} \left[ 1 + \sum_{j \in \partial a \setminus i} \left( \frac{1}{q_{f_{j\rightarrow a}}^{0} + q_{f_{j\rightarrow a}}^{0}} - 1 \right) \right]}. \tag{17}
$$

4. Belief-propagation on factor graphs: theory B

On factor graphs, the original spin-glass picture [4, 11] may still apply, but calls for a modification. Upon the separation of real nodes and factor nodes, belief-propagation (BP) equations are to be distinguished for the two types of nodes on the factor graph. Factor nodes should participate in message-passing, but they should never be selected into FVS, as their removals do not lead to node removal in the original graph. As factor nodes cannot be unoccupied, one of the three cases of a real node’s state is dismissed for factor nodes: the message $q_{i,j}^{0}$ is now 0, if the sender $i$ is a factor node (thus its neighbor $j$ is a real node). Therefore, for a real node $i \in R$, messages are ($\partial x$ denoting neighbors of $x$):

$$
q_{i}^{0} = \frac{1}{z_{i+\partial i}} e^{-\beta} \prod_{a \in \partial i} \left[ \prod_{j \in \partial a \setminus i} (p_{f_{j\rightarrow a}}^{0} + p_{f_{j\rightarrow a}}^{0}) \right],
\tag{18}
$$

$$
q_{i}^{0} = \frac{e^{-\beta} \prod_{b \in \partial(i \cup a)} \left( p_{f_{b\rightarrow a}}^{0} \right) \prod_{a \in \partial i} \left[ \prod_{j \in \partial a \setminus i} (p_{f_{j\rightarrow a}}^{0} + p_{f_{j\rightarrow a}}^{0}) \right]}{z_{i+\partial i}^{R - \partial i}},
$$

$$
q_{i}^{0} = \frac{e^{-\beta}(1 - p_{f_{i\rightarrow a}}^{0}) \prod_{b \in \partial(i \cup a)} \left( p_{f_{b\rightarrow a}}^{0} \right) \prod_{a \in \partial i} \left[ \prod_{j \in \partial a \setminus i} (p_{f_{j\rightarrow a}}^{0} + p_{f_{j\rightarrow a}}^{0}) \right]}{z_{i+\partial i}^{R - \partial i}},
$$

where the normalization constant $z_{i+\partial i}^{R - \partial i}$ allows that $q_{i}^{0} + q_{i}^{0} + \sum_{c \in \partial \cup a} q_{c}^{0} = 1$, and $\beta$ is the inverse temperature controlling the mean number of zero-state nodes in the system. On the other hand, for a factor node $a \in I$, messages are:

$$
P_{a \rightarrow i}^{0} = 0,
\tag{19}
$$

$$
P_{a \rightarrow i}^{0} = \frac{e^{-\beta} \prod_{k \in \partial a \setminus i} \left( q_{k \rightarrow a}^{0} + q_{k \rightarrow a}^{0} \right)}{z_{a+\partial a}^{I - a \cup i}},
$$

$$
P_{a \rightarrow i}^{0} = \frac{e^{-\beta}(1 - q_{f_{a \rightarrow i}}^{0}) \prod_{k \in \partial a \setminus i} \left( q_{f_{k \rightarrow a}}^{0} + q_{f_{k \rightarrow a}}^{0} \right)}{z_{a+\partial a}^{I - a \cup i}},
where $z_{a 
eq i} = 1$ satisfies $p_i^{a} = \sum_{b \in A \setminus i} p_i^{a} \phi_i^a = 1$. The three types of free entropy contribution are now

$$
\langle \phi_i \rangle = \frac{1}{\beta} \ln \left[ 1 + e^{-\beta} \prod_{b \in \partial_i} \left( p_{i-b}^a + p_{b-i}^b \right) + e^{-\beta} \sum_{b \in \partial_i} \left( 1 - p_{i-b}^a \right) \prod_{m \in \partial_i \setminus b} \left( p_{i-m}^a + p_{m-i}^m \right) \right],
$$

$$
\langle \phi_{i-b} \rangle = \frac{1}{\beta} \ln \left[ 0 + e^{-\beta} \prod_{b \in \partial_i} \left( q_{i-b}^b + q_{b-i}^b \right) + e^{-\beta} \sum_{b \in \partial_i} \left( 1 - q_{i-b}^b \right) \prod_{m \in \partial_i \setminus b} \left( q_{i-m}^m + q_{m-i}^m \right) \right],
$$

$$
\langle \phi_{i-b} \rangle = \frac{1}{\beta} \ln \left[ q_{i-b}^b + q_{b-i}^b + (1 - q_{i-b}^b) \left( p_{i-a}^a + p_{a-i}^a \right) \right],
$$

where $z_{a \neq i} \in R$ satisfies $p_i^{a} = \sum_{b \in A \setminus i} p_i^{a} \phi_i^a = 1$. The three types of free entropy contribution are now

from which we calculate the free entropy density $f = \Phi(\beta)/N$. Looking at (18) and (19), one verifies that, for a real node $i$ from a certain clique, messages from other nodes in the same clique are successfully passed to $i$ via the factor node (say $a$) placed in the center of the clique, without adding extra information along the way (since $p_i^{a} = 0$), in which case the original message-passing scheme without the factor node is strictly preserved. This observation also implies that the current factor-node approach would work perfectly when these new nodes are replacing cliques, and may not strictly work along the representation of other factors in the graph (e.g., rectangles, triangles sharing edges etc) that are not complete cliques. If a non-clique factor is represented by a factor node, it will send some extra messages to certain nodes in the factor through channels that do not exist on the original graph.

5. Results

The theories and algorithms are tested on synthetic graphs (random regular triangle/tetrahedron graphs) and then on a panel of real networks of various sorts. Random regular factor graphs are used to demonstrate the matching of theories and numerical outcomes, as well as the binding of analytical lower bounds. Since the graph is regular, considering that messages and cavity probabilities are the same on edges and nodes, BP equations can be simplified and we arrive analytically at the replica-symmetric mean-field theory-A and theory-B. Numerical results are obtained with the original BPD algorithm and the new FBPD and FCoreHD/FCoreHD algorithms are tested on a panel of real networks of various types, including power grid (Grid) [21], internet network at the autonomous system level (IntNet) [22], email networks (Email1, Email2) [23, 24], protein–protein interaction network (yeast) [25], and social networks on multiple platforms (LastFM, Twitch, FB [a/b; from two sources], Deezer, Brightkite [BK]) [26–30]. With each network instance, we consider the identification of cliques up to order $\gamma = 3, 4, 5$; correspondingly, the original graph is transformed to factor graphs at different clique cutoffs (table 1 left). As expected, along an increasing $\gamma$, the factor graph is in general more and more simplified (with some rare exceptions, as discussed in section 1) with smaller $|N|$ and $|E|$. In many cases, $\gamma = 4$ demonstrates the saturation of clique identification as the factor graph is often only slightly changed when we further move to $\gamma = 5$.

The FVS size ($m$) of each network is computed (table 1 right), using BPD/CoreHD on the original graph and FBPD/FCoreHD on the factor graph of different $\gamma$-levels. Consistent with our expectation, with more

5.1. Synthetic graphs

Results of the FVS size of random regular triangle/tetrahedron graphs with different triangle/tetrahedron degree $K$ are shown in figure 4, obtained from the two spin-glass theories, the FBPD algorithm, as well as the lower bound $b^{\text{qua}}(K)$. Unexpectedly, the two spin-glass theories yield identical results; the corresponding two versions of FBPD are further tested on various synthetic and real networks, and all results suggest that the two theories are essentially equivalent, despite built up from distinct thermodynamic pictures. As lower-bound equilibrium solutions, spin-glass theories well match algorithm outcomes, while progressively underestimate numerical results as $K$ gets larger. The lower bound $b^{\text{qua}}(K)$ binds tight at random regular triangle graphs (i.e., $b^{\text{qua}}(K)$), while the binding loosens at random regular tetrahedron graphs (i.e., $b^{\text{qua}}(K)$).

5.2. Real networks

BPD/FBPD and FCoreHD/FCoreHD algorithms are tested on a panel of real networks of various types, including power grid (Grid) [21], internet network at the autonomous system level (IntNet) [22], email networks (Email1, Email2) [23, 24], protein–protein interaction network (yeast) [25], and social networks on multiple platforms (LastFM, Twitch, FB [a/b; from two sources], Deezer, Brightkite [BK]) [26–30]. With each network instance, we consider the identification of cliques up to order $\gamma = 3, 4, 5$; correspondingly, the original graph is transformed to factor graphs at different clique cutoffs (table 1 left). As expected, along an increasing $\gamma$, the factor graph is in general more and more simplified (with some rare exceptions, as discussed in section 1) with smaller $|N|$ and $|E|$. In many cases, $\gamma = 4$ demonstrates the saturation of clique identification as the factor graph is often only slightly changed when we further move to $\gamma = 5$. The FVS size ($m$) of each network is computed (table 1 right), using BPD/CoreHD on the original graph and FBPD/FCoreHD on the factor graph of different $\gamma$-levels. Consistent with our expectation, with more
Figure 4. FVS size of random regular triangle/tetrahedron graphs. Results are obtained from two spin-glass theories (A, B), the FBPD algorithm (averaging over random graph ensembles), as well as the lower bound $b_{tri/qua}$. 

Table 2. Network dismantling results, showing the size of the node removal set ($n$) and the number of re-inserted nodes ($r$). Superscripts $B$, $FB$, $C$, $FC$ are short for BPD, FBPD, CoreHD, FCoreHD; subscripts stand for the values of $\gamma$. Best (smallest) $n$ of each row is shown in bold.

| Network | $n_B$ | $n_{FB}^{10}$ | $n_{FB}^{50}$ | $n_{FB}^{100}$ | $n_C$ | $n_{FC}^{10}$ | $n_{FC}^{50}$ | $n_{FC}^{100}$ | $r_B$ | $r_{FB}^{10}$ | $r_{FB}^{50}$ | $r_{FB}^{100}$ | $r_C$ | $r_{FC}^{10}$ | $r_{FC}^{50}$ | $r_{FC}^{100}$ |
|---------|-------|---------------|---------------|---------------|-------|---------------|---------------|---------------|-------|---------------|---------------|---------------|-------|---------------|---------------|---------------|
| Email1  | 456   | 454           | 455           | 456           | 452   | 458           | 452           | 454           | 15    | 15            | 18            | 14            | 21   | 11            | 22            | 18            |
| Yeast   | 357   | 351           | 354           | 358           | 353   | 355           | 354           | 356           | 36    | 35            | 38            | 28            | 28   | 43            | 34            | 35            |
| FB-a    | 1907  | 1921          | 1894          | 1914          | 1872  | 1872          | 1913          | 543           | 521   | 473           | 542           | 536           | 550  | 479           |                |               |
| Grid    | 312   | 301           | 304           | 307           | 317   | 302           | 305           | 309           | 225   | 174           | 154           | 149           | 191  | 163           | 154           | 144           |
| IntNet  | 160   | 159           | 161           | 160           | 161   | 161           | 156           | 157           | 70    | 51            | 49            | 46            | 67   | 55            | 55            |               |
| LastFM  | 1285  | 1262          | 1258          | 1274          | 1296  | 1296          | 1276          | 1266          | 291   | 234           | 203           | 173           | 294  | 209           | 211           | 203           |
| Twitch  | 3013  | 3011          | 2992          | 2997          | 3036  | 3037          | 3021          | 3035          | 89    | 76            | 73            | 105           | 79   | 86            | 67            |               |
| FB-b    | 3311  | 3295          | 3293          | 3281          | 3356  | 3332          | 3302          | 3327          | 908   | 834           | 805           | 795           | 878  | 839           | 837           | 794           |
| Email2  | 2616  | 2626          | 2647          | 2647          | 2621  | 2648          | 2647          | 2647          | 5244  | 4508          | 5027          | 5027          | 5225 | 4546          | 5021          | 5021          |
| Deezer1 | 16537 | 16415         | 16403         | 16389         | 16891 | 16738         | 16709         | 16729         | 819   | 618           | 552           | 562           | 571  | 623           | 592           | 560           |
| Deezer2 | 23805 | 23751         | 23686         | 23668         | 24180 | 24098         | 24112         | 24065         | 814   | 635           | 654           | 655           | 798  | 667           | 607           | 653           |
| BK      | 6225  | 6140          | 6130          | 6126          | 6257  | 6213          | 6207          | 6197          | 2255  | 1634          | 1497          | 1448          | 1444| 1606          | 1457          | 1410          |

and more cliques being captured (i.e., increasing values of $\gamma$), the FVS size obtained from FBPD gets smaller and smaller until it saturates (table 1 column 9–12), and is the best result (i.e., smallest FVS size) among all candidates. During the experiments, for computational efficiency, after each set of BP iterations we eliminate a handful of nodes (instead of one node) that have the highest probabilities of removal; tests suggest that the performance of FBPD is even more pronounced if we slow down the removal. It should note, however, strictly speaking, the comparison of BPD vs FBPD, CoreHD vs FCoreHD (not including BPD vs CoreHD and FBPD vs FCoreHD) on FVS size is not on a rigorously fair basis, since no loop on the factor graph is not equivalent to no loop on the original graph. Nevertheless, such deviation becomes negligible for real networks with non-trivial clustering, as in practice BPD/CoreHD destroys most local loops.

We dismantle the network such that the LCC size of remaining sub-networks is no more than 1% of the size of the original network. The dismantling set starts with the FVS (of size $m$), experiencing a few tree-breaking steps (of size $t$) that push the subgraphs further down the size threshold, and then a number of nodes (of size $r$) are re-inserted into the remainder graph as much as possible; the final size of the dismantling set $n = m + t - r$. The number of tree-breaking steps $t$ is largely the same across different methods, and is always a small number; results of $n$ and $r$ are shown in table 2. As mentioned earlier, a smaller FVS at network decycling does not always lead to a smaller removal set at network dismantling, because a simpler graph leads to less node re-insertion. This is confirmed in the results (table 2 right): such is the case for the factor graph comparing to the original graph, as well as for the factor graph constructed from a large $\gamma$ comparing to from a small $\gamma$. Indeed, results suggest that although further increasing $\gamma$ leads to a smaller FVS (table 1 right), the best dismantling result is nevertheless often obtained with an intermediate $\gamma$ (table 2 left). Such limitation being said, the improvement of FBPD over BPD and FCoreHD over CoreHD in dismantling performance, is clearly demonstrated at almost all tested instances.
5.3. Scale of graph transformation

As a final note, we analyze the size of the original graph \( G = (N, E) \) and the factor graph \( G' = (N', E') \). Write \( N' = N + \Delta N, E' = E - \Delta E \) (consider the net reduction of edges) where \( E \geq N - 1, E' \geq N' - 1 \), with equality taken when \( G \) or \( G' \) is a tree. Consider the identification of cliques up to order \( \gamma \), withholding the possible existence of incomplete factors (i.e., cliques with certain edges missing). \( G \) is transformed into \( G' \) with factor nodes inserted, where each factor node can have up to \( \gamma \) links. Since an edge in the factor graph is always connecting one real node and one factor node, there is \( N' - 1 \leq E' \leq \gamma \Delta N \), and

\[
1 \geq \frac{\Delta N}{N} \sim \frac{\Delta N}{N - 1} \geq \frac{1}{\gamma - 1}. \tag{21}
\]

The denser the original \( G \) is, the closer \( \Delta N / N \) is to \( 1 / (\gamma - 1) \), i.e., the less are factor nodes added to the graph. Consider a specific complete \( \gamma \)-clique \( s \), with \( e \) edges. After the transformation, there is \( \Delta e_s = \frac{\gamma (\gamma - 1)}{2} - \gamma = \frac{\gamma (\gamma - 3)}{2} \), \( \Delta e_s > 0 \) for \( \gamma > 3 \) and monotonically increases with \( \gamma \). For incomplete cliques, \( \Delta e_s \) is smaller and \( \gamma (\gamma - 3) / 2 \) is the upper bound. The number of cliques identified is the number of factor nodes \( \Delta N \). Thus the largest possible reduction of edges \( \Delta E \) is reached when all cliques are complete \( \gamma \)-cliques, which is also when the lower bound of (21) is reached. Therefore

\[
\Delta E = \{ \Delta e_m \}_{\Delta m = \Delta N, m \in [2, \gamma]} \leq \Delta N \Delta e_s \leq \frac{\gamma (\gamma - 3)}{2} N \sim \frac{\gamma (\gamma - 3) N - 1}{\gamma - 1} \leq \frac{\gamma (\gamma - 3) E}{\gamma - 1},
\]

thus

\[
1 \leq \frac{\Delta E}{E} \leq \frac{\gamma (\gamma - 3)}{2(\gamma - 1)}. \tag{22}
\]

Equations (21) and (22) determine the scale of graph transformation. The upper/lower bound of (21) corresponds to the lower/upper bound of (22). The former case is when the original graph is the simplest (a tree or quasi-tree): in this case, a largest number of factor nodes are added to the original graph, and edge reduction is the smallest (even adding more edges to the graph). The resulting factor graph is then the most unfriendly to the dismantling task. Yet this case is also where the factor-graph approach is the least required, as dismantling could already be well resolved on the original tree-like graph via traditional methods (e.g., BPD). The latter case is when the original graph is most clustered (i.e., full of \( \gamma \)-cliques), but now the number of additional factor nodes is the smallest, and edge reduction is the largest as well. Indeed, this is when we need the new factor-graph solution the most to conduct efficient network dismantling, while the factor graph is now also the most simplified in our favor. This reveals an inherent trade-off in the applicability of BPD vs FBPD (also CoreHD vs FCoreHD): the utility of factor-graph transformation (thus the utility of FBPD/FCoreHD) does not interfere with what we could achieve without this technique, while does desirably providing a family of new solutions (through the manipulation of factors) which essentially span an algorithm spectrum.

6. Discussions and concluding remarks

The re-insertion procedure adopted by many dismantling algorithms [3–5], regarded by many as being ad-hoc, is desirably subdued in scale under the new factor-graph approach (table 2), as a direct result of the preservation of local structures during graph transformation. Essentially, this solution framework allows for the consideration of miscellaneous local structures besides cliques (e.g., squares, triangles sharing edges etc), which are commonly seen in real-world networks especially of certain functional types (e.g., protein interaction networks, network roads). Although the incorporation of incomplete cliques would let the message-passing in the factor graph deviate from that in the original graph (section 4, theory B), this is certainly supported in the new message-passing scheme guided by the novel spin-glass picture (section 3, theory A). It will be interesting to investigate, for example, if network decycling and dismantling results could be further improved and if the re-insertion procedure could be further reduced, although the computational burden could be daunting during such elaboration.

In the experiments we tested a such extension. Instead of identifying small complete cliques as graph factors, we construct larger structures as factors. Specifically, starting from a triangle, we add all neighboring triangles to the structure until no triangle could be added. This agglomerate is then viewed as a single graph factor. Evidently, such a method identifies structures that are much larger in size than the base case, and the graph reduction is pushed to a much higher level where short-range structures of various sorts are largely overlooked (we also tested starting from a tetrahedron and adding neighboring tetrahedra, which leads to slightly smaller structures; the discussion below is similar). Results suggest that, however, this treatment does not lead to smaller dismantling set. The FVS is indeed shrank, as now the factor graph is much more simplified; yet during re-insertion, since now a removed node in FVS often entails large clusters (as now factors are huge), the likelihood of a successful re-insertion, without violating the dismantling size...
constraint, is much lowered. Therefore, with smaller FVS and even smaller re-insertion, the overall result is downgraded.

The current spin-glass theories are not sufficiently binding when the graph is very densely clustered (as for random regular triangle/tetrahedron graphs with a large triangle/tetrahedron degree). In future works, higher-order effects under the replica-symmetry spin-glass theory might be further considered in the model. The binding of mathematical lower bounds of FVS size for random regular factor graphs can certainly be improved as well.

Acknowledgments

The author sincerely thanks Hai-Jun Zhou and Pan Zhang at the Institute of Theoretical Physics, Chinese Academy of Sciences for tremendous hospitality throughout the course of this study. The author thanks the anonymous reviewer and the editor for recognition of this work and the thoughtful feedbacks. The author declares no competing interests.

Data availability statement

All data that support the findings of this study are included within the article (and any supplementary files).

ORCID iDs

Tianyi Li https://orcid.org/0000-0002-4654-9862

References

[1] Chen Y, Paul G, Havlin S, Liljeros F and Stanley H E 2008 Finding a better immunization strategy Phys. Rev. Lett. 101 058701
[2] Morone F and Makse H A 2015 Influence maximization in complex networks through optimal percolation Nature 524 65–8
[3] Braunstein A, Dall’Asta L, Semerjian G and Zdeborová L 2016 Network dismantling Proc. Natl Acad. Sci. USA 113 12368–73
[4] Mugishia S and Zhou H J 2016 Identifying optimal targets of network attack by belief propagation Phys. Rev. E 94 012305
[5] Zdeborová L, Zhang P, Zhou H J and Zhou H J 2016 Fast and simple decycling and dismantling of networks Sci. Rep. 6 37954
[6] Clusella P, Grassberger P, Pérez-Reche F J and Politi A 2016 Immunization and targeted destruction of networks using explosive percolation Phys. Rev. Lett. 117 208301
[7] Wandelt S, Sun X, Feng D, Zanin M and Havlin S 2018 A comparative analysis of approaches to network-dismantling Sci. Rep. 8 1–15
[8] Ren X-L, Gleinig N, Helbing D and Antulov-Fantulin N 2019 Generalized network dismantling Proc. Natl Acad. Sci. USA 116 6554–9
[9] Fan C, Zeng L, Sun Y and Liu Y Y 2020 Finding key players in complex networks through deep reinforcement learning Nat. Mach. Intelligence 1–8
[10] Fan C, Zeng L, Feng Y, Cheng G, Huang J and Liu Z 2020 A novel learning-based approach for efficient dismantling of networks Int. J. Mach. Learn. Cybern. 1–11
[11] Zhou H J 2013 Spin glass approach to the feedback vertex set problem Eur. Phys. J. B 86 455
[12] Sun J, Liu R, Fan Z, Xie J, Ma X and Hu Y 2018 Lower bound of network dismantling problem Chaos 28 063128
[13] Wang Z, Sun C, Yuan G, Rui X and Yang X 2020 A neighborhood link sensitive dismantling method for social networks J. Comput. Sci. 43 101129
[14] Zhao D, Yang S, Han X, Zhang S and Wang Z 2020 Dismantling and vertex cover of network through message passing IEEE Trans. Circ. Syst. II 67 2732
[15] Li T, Zhang P and Zhou H-J 2021 Long-loop feedback vertex set and dismantling on bipartite factor graphs Phys. Rev. E 103 L061302
[16] Karrer B and Newman M E 2010 Random graphs containing arbitrary distributions of subgraphs Phys. Rev. E 82 066118
[17] Yoon S, Goltsvev A V, Dorogovtsev S N and Mendes J F F 2011 Belief-propagation algorithm and the Ising model on networks with arbitrary distributions of motifs Phys. Rev. E 84 041144
[18] Miller J C 2009 Percolation and epidemics in random clustered networks Phys. Rev. E 80 020901
[19] Newman M E 2009 Random graphs with clustering Phys. Rev. Lett. 103 058701
[20] Bau S, Wormald N C and Zhou S 2002 Decycling numbers of random regular graphs Random Struct. Alg. 21 397–413
[21] Watts D J and Strogatz S H 1998 Collective dynamics of ‘small world’ networks Nature 393 440
[22] Leskovec J, Kleinberg J and Faloutsos C 2005 Graphs over time: Densification laws, shrinking diameters and possible explanations in Proc. of the 11th ACM SIGKDD Int. Conf. on Knowledge Discovery in Data Mining pp 177–87
[23] Leskovec J, Kleinberg J and Faloutsos C 2007 Graphs over time: Densification laws, shrinking diameters and possible explanations ACM Trans. Knowl. Discov. Data 1 2–es
[24] Leskovec J, Lang K J, Dasgupta A and Mahoney M W 2009 Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters Int. Math. 6 29
[25] Bu D et al 2003 Topological structure analysis of the protein–protein interaction network in budding yeast Nucl. Acids Res. 31 2443–50
[26] Rozemberczki B and Sarkar R 2020 Characteristic functions on graphs: Birds of a feather, from statistical descriptors to parametric models in Proc. of the 29th ACM Int. Conf. on Information and Knowledge Management pp 1325–34
[27] Rozemberczki B, Allen C and Sarkar R 2021 Multi-scale attributed node embedding *J. Complex Netw.* 9 cnab014
[28] McAuley J and Leskovec J 2012 Learning to discover social circles in ego networks *NIPS* 1 548
[29] Rozemberczki B, Davies R, Sarkar R and Sutton C 2019 Gemsec: Graph embedding with self clustering in *Proc. of the 2019 IEEE/ACM Int. Conf. on Advances in Social Networks Analysis and Mining* pp 65–72
[30] Cho E, Myers S A and Leskovec J 2011 Friendship and mobility: user movement in location-based social networks in *Proc. of the 17th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining* pp 1082–90