This is an author produced version of Robustness analysis of network modularity.

White Rose Research Online URL for this paper:
http://eprints.whiterose.ac.uk/88836/

Article:

https://doi.org/10.1109/TCNS.2015.2476197

© 2015, IEEE. Uploaded in accordance with the publisher's self-archiving policy. Personal use of this material is permitted. Permission from IEEE must be obtained for all other users, including reprinting/ republishing this material for advertising or promotional purposes, creating new collective works for resale or redistribution to servers or lists, or reuse of any copyrighted components of this work in other works.
Robustness analysis of network modularity

Jongrae Kim, Member, IEEE, and Kwang-Hyun Cho, Senior Member, IEEE

Abstract—Modules are commonly observed functional units in large-scale networks and the dynamics of networks are closely related to the organization of such modules. Modularity analysis has been widely used to investigate the organizing principle of complex networks. The information about network topology needed for such modularity analysis is, however, not complete in many real-world networks. We noted that network structure is often reconstructed based on partial observation and therefore it is re-organized as more information is collected. Hence, it is critical to evaluate the robustness of network modules with respect to uncertainties. For this purpose, we have developed a robustness bounds algorithm that provides an estimation of the unknown minimal perturbation, which breaks down the original modularity. The proposed algorithm is computationally efficient and provides valuable information about the robustness of modularity for large-scale network analysis.

Index Terms—Network modularity, community structure, robustness analysis

I. INTRODUCTION

Network or graph theory has been applied to modelling many physical, biological, and social systems for various interaction data such as internet communications, biomolecular interactome, and social relationships. A network consists of nodes and edges as shown in Fig. 1(a), where a node may represent a computer server in the internet, a protein species in a protein-protein interaction network, or an individual person in a social network, and an edge may denote a physical network connection between two computers, a protein-protein interaction between the protein species, or friendship between two people. Because of the simplicity of network modelling, a massive number of components and interactions can be considered easily for many cases.

The most important finding in large-scale network analysis is arguably the scale-free characteristic [1]. This explains two important properties, i.e., robustness and small-worldness, in large-scale networks. Another important way of comprehending large-scale networks is modularity analysis, which has been one of academic research interest in recent years. There are several different definitions on network modularity [2], [3], [4], [5], [6]. Among these, a defining characteristic of a module is that nodes in the same module have more frequent interconnections than to the connections to the nodes in different modules. The formulation proposed by Newman [2] is one of the widely accepted definitions as it shows a quite intuitive result and the module calculation can be done efficiently using the power iteration. The community or modular structure provides us with the information about the hidden functional organization of the networks. For instance, two modules indicated by the ellipsoids in Fig. 1(a) indicate that social division occurred in a Karate club in America [7], where the network shows the friend-relationships among the club members.

A profound consequence of the modular structure of complex networks is the enhanced robustness to various internal and external perturbations and disturbances. Robustness is considered to be one of the key factors that shaped biological systems through evolution. Modular system design is an efficient way to distribute and organize functions as frequently observed in many engineering systems, whose design evolves as well based on their performance. The functional modularization might be the origin of robustness [8] and highly optimized tolerance [9]. In addition, graph partition is an important control problem to organize multiple agents in order to perform a common mission while communications among them are limited [10].

A number of previous studies reported how to dissect hierarchical modular structures [11] and interpret their physical, biological and social meanings [11], [12].

However, in many cases, it was overlooked that most large-scale network data are incomplete and that they are only partial measurements of the unknown full networks and/or
a snap shot at a fixed time. For instance, we may not have
the full network data as shown in Fig. 1(a) but only have the
partial sampling such as Fig. 1(b) or 1(c). As the available
network data are only a partial subset of the unknown true
network, the modular structure inferred from such data would
be influenced by the sampling effect as illustrated in Fig. 1(b)
where one node is included in a wrong module. In addition, a
sampled network might include a false interaction, e.g. the
gray edge in Fig. 1(c) (false positive) or miss a true edge
between one of the blue nodes and the lighter blue nodes (false
negative). This sampling effect was reported in the past. For
e.g., identifying high degree nodes in different categories
of biological networks [11] cannot be supported from the data
used [15] and the power-law degree distribution in scale-free
networks is highly sensitive to the data analyzed [14]. Hence,
any network modularity analysis needs to be further validated
by robustness analysis with respect to the network uncertainty
in terms of false positive or negative nodes and edges.

To examine the effect of such uncertainties on the modularity
structure, we need to identify the minimal perturbation that
can break down the original modularity of the network. For
instance, a simple six-nodes network shown in Fig. 2 can be
divided into two modules, the red and the blue. By applying
all possible perturbations, we find that removing three edges
shown in Fig. 2 is the minimum number of edge perturbations,
which destroys the original modularity. Based on this minimal
perturbation, we can measure the robustness of the current
modular structure. The number of possible perturbations to
be examined for an exhaustive search increases exponentially
along with the size of a network and therefore it is impossible
to perform a full search even for a network of a moderate size.

This paper is organized as follows. First, the robustness
analysis is formulated as a quadratic integer programming
problem. Second, the upper and lower bound algorithms are
established. Third, the algorithms are applied to various ex-
ample networks including a social network, the yeast protein-
protein interaction (PPI) network, and a research citation
network. Finally, conclusions are made.

II. ROBUSTNESS OF MODULARITY

An \( n \times n \) adjacency matrix, \( A \), describes a network with \( n \)
number of nodes, where the \( i \)-th row and \( j \)-th column of the
matrix \( A \) is set to 1 if the two nodes are directly connected or
0 if there is no direct connection. The solution of the following
maximization problem \([2]\):

\[
\text{Maximize } Q(s, A) := \frac{1}{4m} s^T B s,
\]

divides \( n \) nodes in \( A \) into two groups for \( Q > 0 \) or declares
the network indivisible for \( Q \leq 0 \), where \( S \) is the set of \( n \-
dimensional column vectors, \( s \), whose element is either 1 or -1,
\( m \) is the number of edges in the network, \( s^T \) is the transpose,
\( k = A1 \), each value in \( k \) is called the degree of node, \( 1 \) is the
\( n \)-dimensional column vector whose elements are all 1, and
\( B := A - \frac{kk^T}{2m} \).

\( B \) measures the difference between the current edge distri-
bution, \( A \), and the average edge distribution, \( kk^T/(2m) \). The
maximum value of \( Q \) being positive indicates more edges than
expected in each subgroup for a division given by \( s \), and the
nodes are separated into two groups depending on the sign of
elements in \( s \).

With the optimal solution to \((1)\) denoted by \( s^* \), the maxi-
mum modularity, \( Q^* \), is given by

\[ Q^* = \max(Q) = Q(s^*, A). \]

While \( A \) is fixed in the maximisation problem, in reality, the
network is most likely a subset of the unknown true network
including some false positive or false negative edges/nodes,
and it might even change with time. For brevity, only the edge
perturbation case is considered and the general case including
node perturbation will be discussed at the end. Once edges are
added to and/or removed from the current network, the
adjacency matrix is changed.

\[ A_g := A + \Delta_A, \]

where the subscript \( g \) represents the perturbed network, \( \Delta_A \)
is \( n \times n \) matrix representing removal (-1) or addition (+1) of
edges to the original network. The perturbed \( B \) is given by

\[ B_g := A_g - \frac{1}{2m_g} k_g k_g^T, \]

\[ k_g := A_g 1 = k + \delta_k, \]

\( m_g \) is the number of edges in the perturbed network, \( 1 \) is
assumed to have an appropriate dimension from now on, and
\( \delta_k \) is an \( n \)-dimensional vector, whose elements represent
the degree changes of the nodes in the network. he robustness
analysis problem is formulated as follows:

\[ \text{Problem 1: (Robustness analysis of modularity) For a given}
\text{network, } A, \text{ the partition, } s^*, \text{ find } \Delta_A \text{ minimising } Q_g \text{ as}
\text{follows:} \]

\[
\text{Minimize } Q_g(s^*, \Delta_A) \quad \Delta_A
\]

for a fixed number of alterations, \( t \in [1, \min(t_1, t_2)] \), where
\( Q_g(s^*, \Delta_A) := Q(s^*, A_g), t_1 = m \) and \( t_2 = n(n-1)/2 - m \).

For each number of alterations, \( t \), the worst perturbation, \( \Delta_A \),
to impact on the modular structures of \( A \) is to be sought.
There exist always two extreme perturbations: removing all
\( m \) original edges and all nodes in \( A \) become orphan; or
connecting each node to the other nodes and \( A \) is fully
connected. The upper bound of $t$ corresponds to either one of these two extreme cases. It can be shown that the following is equivalent to Problem 1:

**Problem 2:** (Robustness analysis of modularity) For $t$ in the range of $[1, \min(t_1, t_2)]$, find $d_v$ such that

$$\text{Minimize } q(d_v) = a \cdot d_v - \frac{(b \cdot d_v)^2}{b},$$

where $D_v$ is the set of all feasible column vectors, $d_v$, whose dimension is $n(n-1)/2$ and the value of each element is 0 (no change) or 1 (either remove the edge if an edge exists or add an edge if not). $d_v$ is constructed by vectorizing $\Delta_A$ and $d_v^T 1 = t$. “.” is the dot product, $a$ and $b$ are vectors, which are constructed from $A$, $m$, and $s^*$, and $b$ is the magnitude of $b$ (see Appendix for the full definitions).

**Proof:** See Appendix.

Once the minimization problem is solved, the worst $Q_g$ is calculated as follows:

$$Q_g^{\text{wast}}(t) := \min_{\alpha \in S_\alpha(t)} Q_g$$

$$= \min_{\alpha \in S_\alpha(t)} \left\{ \frac{1}{1 + \alpha} \left[ Q^* + \frac{\alpha (k \cdot s^*)^2}{8m^2(1 + \alpha)} + \frac{q^*}{4m} \right] \right\},$$

where $q^*$ is the minimum of $q(d_v)$, $\alpha$ is given by

$$2m_g = 1^T A_g 1 = 2m(1 + \alpha),$$

and $S_\alpha(t)$ is the set of all possible elements of $\alpha$ for a fixed $t$ as follows:

$$S_\alpha(t) = \begin{cases} \{0, \pm 2/m, \pm 4/m, \ldots, \pm t/m\} & \text{for } t \text{ even}, \\ \{\pm 1/m, \pm 3/m, \ldots, \pm t/m\} & \text{for } t \text{ odd}. \end{cases}$$

$\alpha$ is the number of edge alterations. Positive or negative values of $\alpha$ imply that after perturbation the number of edges in $A$ has increased or decreased, respectively. For a fixed number of alterations, $t$, there is more than one possible value of $\alpha$ given by the set $S_\alpha(t)$.

Modularity robustness analysis is presented as a quadratic integer programming problem. The computational cost increases exponentially as fast as $\sum_{k=1}^{n} n!/[k!(n-k)!]$. Calculating the exact solution requires unreasonable computation time for even some moderate size problems. Hence, developing an efficient lower and upper bounds algorithm is greatly desirable. However, we note that any bounds algorithm will eventually produce conservative results for some cases, which is the unavoidable risk for using bounds algorithms.

**A. Robustness lower bound**

By the definition of vector dot product, the minimization problem, (2), can be written as

$$\text{Minimize } q(d_v) = ad_v \cos \theta_1 - bd_v^2 \cos^2 \theta_2$$

subject to $d_v \cdot 1 = t$, where $t \in [1, \min(t_1, t_2)]$, $a$ and $d_v$ is the magnitude of $a$ and $d_v$, respectively. The angle between $a$ and $d_v$ is $\theta_1$, while the angle between $b$ and $d_v$ is $\theta_2$. It can be shown that $\theta_1$ is in the following range:

$$\cos^{-1} \left( \frac{\sum_{i \in M} a_i}{a\sqrt{t}} \right) \leq \theta_1 \leq \cos^{-1} \left( \frac{\sum_{i \in M} a_i}{a\sqrt{t}} \right),$$

where $M$ and $\overline{M}$ are the sets, whose elements are the indices of the first $t$-number of largest and smallest elements in $a$, respectively. $\theta_2$ is equal to $\pi + \theta - \theta_1$ for $\theta + \theta_1 + \theta_2 > \pi$ or $\pi - \theta - \theta_1$ otherwise (See Proposition [A.1] in appendix). The minimizing $q(d_v^*)$ is shown to be equivalent to:

$$\text{Minimize } q(\theta_1) = a\sqrt{t}x - bt(x \cos \theta \pm \sqrt{1 - x^2 \sin^2 \theta})^2,$$

and the minimum of $q(\theta_1)$ occurs at $x^*$, which is either the solution of quartic equation, i.e., $\sum_{i=0}^{4} w_i x^i = 0$, where $x = \cos \theta_1$, or one of the boundary values for $\theta_1$, i.e., $x = \cos \theta^*_1$ or $x = \cos \theta^*_1$. The definitions of $w_i$ and the proofs are shown in Propositions [A.2] and [A.3] in appendix. All solutions of the quartic equations for $x$ can easily be calculated and the minimum solution, $\theta^*_1$, is given by $\cos^{-1} x^*$. Now, we are ready to present a lower bound algorithm.

**Theorem 2.1:** (Lower Bound) For a given $t$, the worst case, $Q_g^{\text{wast}}(t)$, is bounded below by

$$Q_{LB}[\alpha_{LB}(t)] \leq Q_g^{\text{wast}}(t),$$

where $\alpha \in S_\alpha(t)$,

$$Q_{LB}(\alpha) := \frac{1}{1 + \alpha} \left[ Q^* + \frac{\alpha (k \cdot s^*)^2}{8m^2(1 + \alpha)} + \frac{q(\theta_1)}{4m} \right],$$

$$\alpha_{LB}(t) = \arg\min_{\alpha \in S_\alpha(t)} Q_{LB}(\alpha).$$

**Proof:** By the definition, $q(\theta^*_1)$ is less than or equal to $q^*$, and it leads to $Q_{LB}[\alpha_{LB}(t)] \leq Q_g^{\text{wast}}(\alpha)$. ■

In order to find the lower bound, first, calculate $\min q(\theta_1)$ for all $\alpha \in S_\alpha(t)$, second, substitute these into $Q_{LB}(\alpha)$, take the minimum among $Q_{LB}(\alpha)$ for $\alpha \in S_\alpha(t)$, and finally, repeat these for different $t$ values. This algorithm requires only polynomial computation time.

**B. Robustness upper bound**

Whether the lower bound is close to the true worst or not can be verified by an upper bound. To develop an upper bound, the following inequality is derived:

$$\min_{d_v \in D_v} q(d_v) \leq q(d_v^*),$$

where $d_v$ represents some specific perturbation, $\Delta_A$, defined by Proposition [A.4] in appendix. The next step is to solve the following minimization problem, which is constructed from $q(d_v)$ shown in Proposition [A.4].

$$\text{Minimize } p(d_v) = (a^T - a^T) A_v d_v - d^T \bar{b} b^T d_v.$$
This is only a function of $d_v$ excluding $\alpha$. Expand the vector multiplications,

$$
p(d_v) = a_1d_{v1} + a_2d_{v2} + \ldots + a_l d_{vl} - \left( b_1d_{v1} + b_2d_{v2} + \ldots + b_l d_{vl} \right)^2,
$$

where $a_i$, $b_i$, and $d_{vi}$ are the $i$-th element of $(a^T - \tilde{a}^T)A_v$, $b$ and $d_v$, respectively, for $i = 1, 2, \ldots, l - 1, l$, and $l = n(n-1)/2$. Notice that $d_{vi}^2 = d_{vi}$ as $d_{vi}$ is either 0 or 1.

For brevity, consider $n = 3$ case, the formulations for the general cases can be derived similarly.

$$
p(d_v) = c_1d_{v1} + c_2d_{v2} + c_3d_{v3} - 2b_1b_2d_{v1}d_{v2} - 2b_1b_3d_{v1}d_{v3} - 2b_2b_3d_{v2}d_{v3},
$$

where $c_i = a_i - \tilde{a}_i^2$ for $i = 1, 2, 3$. Again, this is a quadratic integer programming problem. Although any perturbation will provide an upper bound, in order to reduce the unknown distance from the worst case and simplify the calculations, $p(d_v)$ is modified as follows:

$$
\bar{p}(d_v) = c_1d_{v1}d_{v2} + c_2d_{v1}d_{v3} + c_3d_{v2}d_{v3} - 2b_1b_2d_{v1}d_{v2} - 2b_1b_3d_{v1}d_{v3} - 2b_2b_3d_{v2}d_{v3},
$$

i.e.,

$$
\bar{p}(d_v) = f^T \tilde{d}_v,
$$

where

$$
f^T := \begin{bmatrix} c_1 + c_2 - 2\tilde{b}_1\tilde{b}_2 c_1 + c_3 - 2\tilde{b}_1\tilde{b}_3 c_2 + c_3 - 2\tilde{b}_2\tilde{b}_3 \end{bmatrix},
$$

$$
\tilde{d}_v := \begin{bmatrix} d_{v1}d_{v2} & d_{v1}d_{v3} & d_{v2}d_{v3} \end{bmatrix}^T \in \mathbb{D}_{v}. \]

The minimum value $\bar{p}(d_v)$ is obtained by simply choosing the first $\tau$ smallest elements in $f$ and set the corresponding elements of $\tilde{d}_v$ to 1, where $\tau$ is an integer in $[1, l]$. This is a heuristic modification of $p(d_v)$. There is no guarantee that a minimising solution of $\bar{p}(d_v)$ is the same as the one of $p(d_v)$. This is the reason that the solution for $\bar{p}(d_v)$ will be an upper bound, where calculating the solution for the modified equation is simply a sorting procedure.

The following inequality is obtained using the solution obtained from $\bar{p}(d_v)$:

$$
q(\tilde{d}_v) \leq q(d_v),
$$

where $\tilde{d}_v$ is a specific perturbation calculated from the solution of $\bar{p}(d_v)$. A detailed proof is shown in Proposition A.5 in appendix.

Now, the upper bound is given by the following Theorem 2.2.

**Theorem 2.2:** (Upper Bound) For a given $t$, the worst case perturbation is bounded above by

$$
Q_{UB}(t) := \frac{1}{1 + \bar{\alpha}} \left[ Q^* \right. + \frac{\bar{\alpha}(k \cdot s^*)^2}{8m^2(1 + \bar{\alpha})} + \frac{q(d_v)}{4m} \left. \right],
$$

for the right hand side of the equation less than $Q^*$ or $Q_{UB}(t) = Q^*$ otherwise, where $\bar{\alpha} = 1 - A_v d_v$. Proof: The proof is trivial and omitted. ■

In the upper bound calculation, the perturbed modularity is compared with the nominal modularity. This is to ensure that the upper bound is always below $Q^*$. The upper bound calculation does not guarantee that the perturbation will always decrease the modularity. The perturbation calculated by the algorithm might improve the modularity of original network by chance and the perturbed modularity will be larger than $Q^*$. For these rare cases, the calculated upper bound will be rejected and the unperturbed one is declared as the upper bound.

In order to improve the upper bounds, some heuristic optimization algorithms could be used such as genetic algorithms, particle swarm optimization, and simulation annealing, where the estimate provided by the upper bound algorithm could be an initial guess.

**C. Subnetwork robustness bounds**

Once a given network is divided into two modules, each module is investigated again whether it can be further divided or not and this procedure is repeated until all modules are no longer divisible. The minimization problem for subnetwork modularity robustness is given by Theorem 2.3.

**Theorem 2.3:** (Subnetwork Robustness) The minimization sub-problem for the worst case analysis of subnetwork is

$$
\begin{align*}
&\text{Minimize} & q^{SG}(d_v) = a \cdot d_v - \frac{(b \cdot d_v)^2}{b} \\
& & + 2m\alpha^sg + \frac{(m^sg + m\alpha^sg)^2}{m(1 + \alpha^sg)},
\end{align*}
$$

where $\alpha^sg$, $m^sg$, $a$, $b$, and all other notations follow similar definitions of the full network.

Proof: See the appendix. ■

The minimization problem for subnetwork robustness is exactly the same as the previous minimization problem except the last two constant terms in (4), which does not affect the minimization solution. Hence, the same lower and upper bounds algorithms for the full network are used for the subnetwork robustness analysis.

**III. Examples**

The bound algorithms are applied to various examples: social, biological, and citation networks. Several physical and biological interpretations are presented.
A. A simple network

The network shown in Fig. 2 has six nodes and seven edges. The two modules, red and blue, are the optimal partition. The upper and lower robustness bounds are illustrated in Fig. 3. The true worst perturbation found by an exhaustive search is indicated in the black circled line. The upper bound presents the worst case perturbation scenario and \( t = 0 \) corresponds to the original network without any perturbation. The first negative value corresponds to the smallest number of perturbations that make the original two module partition invalid. The perturbed network in Fig. 2 shows the worst case perturbation. After removing the three edges, one module disappears and this leaves only the blue module with an additional node that originally belongs to the red module. The lower bound shows that the modularity measure will be negative for the three perturbations. Note that the negative modularity implies that the original partition is destroyed. The robustness of the network module is measured as 43% (addition/removal of three edges out of seven edges) where the upper and lower bounds become negative at the same level of perturbations, i.e., \( t = 3 \).

B. Karate network

The robustness analysis result of the Karate network is shown in Fig. 4. This Karate network illustrates the actual social division that took place among people in a Karate Club in America in 1970’s where each node represents an individual member and each edge denotes the relationship between two members in the club [7]. From the robustness analysis of this division, we found that such division can hold up to 16% perturbations \( (t/m) \) before the lower bound becomes negative. An exhaustive search is not possible for this network since there are too many combinations. The minimum worst change \( (t/m) \) found in order to resolve the social division is 42% perturbation. This implies that if a perturbation corresponding to this upper bound is applied so that some relations are prohibited and new connections are encouraged, the social division might be resolved.

C. Yeast protein-protein interaction network

The protein-protein interaction (PPI) network of yeast is a well-characterized biological interaction network [15]. Each node in this network represents a particular protein and each edge connecting two proteins indicates an identified biomolecular interaction between them. The network has several isolated groups and the largest one composed of 1,004 nodes and 8,319 edges and is used in this analysis. The worst lower bound shown in Fig. 5 is 2% and this indicates that we might have a very conservative lower bound, which is not close to the worst upper bound, 34% perturbation. It might be the opposite case where the upper bound is conservative and the lower bound indeed indicates the extreme fragility of the network modularity structure. This is an unavoidable result in any bounding algorithms corresponding to an NP-hard problem.

D. Citation Network

Due to limitations of the current social network database and measurement technologies for biological networks, time-series data for network growth is still rarely recorded. One available case is the citation network of High-Energy Physics Theory in
The scale-free network since the scale-free network always organises with some narrow concentrated topics. This is completely opposite to the modularity dynamics of an academic society. The citation network and the scale-free network are distributed optimally to the existing two modules by maximising the modularity, i.e., the worst bounds for the current module are calculated until the module is broken down. Once it is broken down, then a new modular structure is found and repeat the calculation.

The modularity robustness analysis is performed as follows: i) current network is divided into two modules, ii) the worst upper ($\bar{\mu}_m$) and lower ($\bar{\mu}_m$) bounds are calculated using the bounds algorithms, iii) once additional nodes with connections to the existing nodes are introduced, the additional nodes are distributed optimally to the existing two modules by maximising the modularity, $Q$, iv) if the modularity is negative, then we go to step i), otherwise we go to step ii) with the updated network by the additional nodes and edges. In other words, the worst bounds for the current module are calculated until the module is broken down. Once it is broken down, then a new modular structure is found and repeat the calculation.

The number of increasing nodes is roughly the same for both networks. Fig. 6 shows the worst bounds histories for both networks. The gap between the bounds for the scale-free network becomes larger as time evolves and the initial modular structure remains the same. The increasing gap with time is mainly caused by the conservatism of the lower bound calculation. On the other hand, the lower bound for the citation network is not conservative and the gap between them is very small once in a while, which implies there is a highly dynamic mixing nature of the citation modularity. The citation modules are not fixed but there exists a strong mixing and re-organising force in the network, which seems quite normal in an academic society with some narrow concentrated topics. This is completely opposite to the modularity dynamics of the scale-free network since the scale-free network always maintains the original modular structure. In other networks, these mixing forces and the modularity conservation energy might be balanced in some ways.

IV. CONCLUSIONS & FUTURE WORKS

An efficient algorithm for the robustness analysis of network modularity is developed. The algorithm calculates the lower and upper bounds of robustness with respect to structural perturbation of the network. The computational cost does not increase exponentially with the number of nodes. Hence, the bounds for a time-varying network, i.e., nodes alterations, can be obtained by applying the algorithm for each fixed time without incurring a significant computational cost.

The tightness of the bounds is case dependent. Some optimization algorithms can be further employed to obtain a tighter bound with the cost of increasing computational time. In general, however, the modular structure starts breaking down from the submodules, which have a smaller number of nodes. In most cases we are more interested in the robustness analysis of small to medium size networks. Therefore, the proposed algorithms can provide valuable information on the fundamental robustness nature of modular structures of complex networks in many practical cases.

The bound estimation algorithms assume that a modular partition, which might not be optimal, is provided based on the modularity definition. As long as the partition is not significantly different from the true, it is unlikely that the worst perturbation would enhance the true partition. However, there are several degeneracy cases for finding the community structures by maximizing the modularity as shown in [18]. Whenever the robustness analysis shows that a network module is fragile, then the modularity partition should be re-investigated whether there exists a better partition.

As one of the important future works, network perturbations corresponding to minimizing or maximising the modularity could be identified as malicious attacks to the network or defence mechanisms of the network. This leads to a min-max optimization problem and it would be one of the ways to design robust network structure with respect to external disturbances.

ACKNOWLEDGEMENT

This work was initiated by the exchange visiting between J. Kim and K.-H. Cho. J. Kim acknowledges the support from the EPSRC grant, EP/G036195/1. K.-H. Cho acknowledges the support from the National Research Foundation of Korea (NRF) grants funded by the Korea Government, the Ministry of Science, ICT & Future Planning (2014R1A2A1A10052404 and 2013M3A9A7046303), the KAIST Future Systems Healthcare Project from the Ministry of Science, ICT & Future Planning, the 2015 GIST Systems Biology Infrastructure Establishment Grant, the KUSTAR-KAIST Institute, Korea under the R&D program supervised by the KAIST, and the grant of the Korean Health Technology R&D Project, Ministry of Health & Welfare, Republic of Korea. (HI13C2162). The authors also thank Dr Nicholas Raske at University of Leeds for proof reading.
APPENDIX

DERIVATION OF 2

Expand $Q_g$ as follows:

$$
\begin{align*}
\text{Minimize } Q_g(s^*, \Delta_A) &= \frac{1}{1 + \alpha} \left\{ Q^* + \frac{1}{4m} \left[ s^{*T} \Delta_{11}s^* - \frac{1}{2m(1 + \alpha)} \right. \right. \\
&\left. \left. \times \left( 2s^{*T} k \delta_k^T s^* + s^{*T} \delta_k^T s^* - \alpha s^{*T} kk^T s^* \right) \right] \right\}.
\end{align*}
$$

For a fixed $\alpha$, the minimisation problem is reduced to

$$
\begin{align*}
\text{Minimize } q(\Delta_{11}) := s^{*T} \Delta_{11}s^* - \frac{1}{2m(1 + \alpha)} \left( 2s^{*T} k \delta_k^T s^* + s^{*T} \delta_k^T s^* \right).
\end{align*}
$$

Re-arrange

$$
\Delta_{11}s^* = \begin{bmatrix} d_1^T \\ d_2^T \\ \vdots \\ d_n^T \end{bmatrix} s^* = \begin{bmatrix} (I_n \otimes s^T) d_1 \\ (I_n \otimes s^T) d_2 \\ \vdots \\ (I_n \otimes s^T) d_n \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_n \end{bmatrix},
$$

where $d_i^T$ is the $i$-th row of $\Delta_{11}$, $I_n$ is an $n \times n$ identity matrix, and $\otimes$ is the Kronecker product. As $\Delta_{11}$ is a symmetric matrix and $n^2$ elements of $d_i$ for $i = 1, 2, \ldots, n$ are not completely independent but only $n(n - 1)/2$ elements are independent. By defining a matrix $L$ appropriately, the following can be found:

$$
\begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_n \end{bmatrix} := L \begin{bmatrix} d_{1,1}^{2,n} \\ d_{1,2}^{2,n} \\ \vdots \\ d_{n,1}^{2,n} \end{bmatrix} = L \tilde{d}_v,
$$

where $d_{i,j}^{1,n}$ is the vector only taking the elements from $j$-th to $n$-th elements of $d_i$ for $i = 1, 2, \ldots, n$ and $j = 2, 3, \ldots, n - 1$. In addition, each element of $\tilde{d}_v$ cannot be freely +1 (add edges) or -1 (remove edges) but can only be +1 or -1 if the corresponding element of $A$ is 0 (no edge) or 1 (pre-existing edge). In order to restrict each element of $\tilde{d}_v$ to 0 (no change) or 1 (change: remove the edge if there is an edge or add an edge if there is no edge) without considering the corresponding element value of $A$, define a diagonal matrix, $A_v$, composed from the element of $A$, i.e., $a_{ij}$.

$$
A_v := \text{diag} \left( f(a_{12}), f(a_{13}), \ldots, f(a_{1n}), f(a_{23}), f(a_{24}), \ldots, f(a_{2n}), \ldots, f(a_{(n-2)(n-1)}), f(a_{(n-2)n}), f(a_{(n-1)n}) \right),
$$

where $f(a_{ij})$ is equal to -1 for $a_{ij} = 1$ or 1 for $a_{ij} = 0$, for $i = 1, 2, \ldots, n - 1$ and $j = 2, 3, \ldots, n$. Then,

$$
\begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_n \end{bmatrix} = L \tilde{d}_v := L A_v d_v,
$$

where $\bar{d}_v$ is the element of $D_v$ and $D_v$ is the set of $n(n - 1)/2$ dimensional vectors, whose element is either 0 or 1. Hence,

$$
\Delta_{11}s^* = (I_n \otimes s^T) LA_v d_v
$$

and

$$
\delta_k = \Delta_{11} \mathbf{1} = (I_n \otimes \mathbf{1}^T) LA_v d_v.
$$

Finally, the minimization problem is reposed as follows:

$$
\begin{align*}
\text{Minimize } q(d_v) &= a^T d_v - d_v^T B d_v, \quad \text{for } d_v \in D_v,
\end{align*}
$$

where

$$
\begin{align*}
\alpha := \left( s^{*T} (I_n \otimes s^T) L - \frac{s^{*T} k s^T (I_n \otimes \mathbf{1}^T) L}{m(1 + \alpha)} \right) A_v, \\
B := \left( A_v^T L^T (I_n \otimes \mathbf{1}^T) s^{*T} (I_n \otimes \mathbf{1}^T) L A_v \right) \frac{2m(1 + \alpha)}{m(1 + \alpha)}.
\end{align*}
$$

As $B$ is a rank one matrix,

$$
\begin{align*}
\text{Minimize } q(d_v) &= a^T d_v - d_v^T B d_v, \quad \text{for } d_v \in D_v,
\end{align*}
$$

where $B = \mathbf{b e}^T$, each element in $\mathbf{b}$ is the magnitude of each row vector of $B$ and $\mathbf{e}$ is the unit vector spanning the one-dimensional row space of $B$. Note that $B$ is symmetric and $\mathbf{b}$ and $\mathbf{e}$ are parallel. Hence, 2 is obtained.

INEQUALITY FOR $\theta_1$

**Proposition A.1**: $\theta_1$ and $\theta_2$ are related to each other as $\theta_2 = \pi + \theta - \theta_1$ for $\theta + \theta_1 + \theta_2 > \pi$ or $\theta_2 = \pi - \theta - \theta_1$ otherwise, where $\theta$ is the angle between $\mathbf{a}$ and $-\mathbf{b}$. $\theta_1$ is in the range between $\theta_1$ and $\bar{\theta}_1$, where

$$
\bar{\theta}_1 := \min(\theta_1) = \cos^{-1} \left( \frac{\sum_{i \in M} a_i}{a \sqrt{M}} \right),
$$

which is greater than or equal to zero, $M$ is the index set whose elements are the indices of the first $t$-number of largest elements in $\mathbf{a}$,

$$
\bar{\theta}_1 := \max(\theta_1) = \cos^{-1} \left( \frac{\sum_{i \in M} a_i}{a \sqrt{M}} \right),
$$

which is less than or equal to $\pi$, and $M$ is the index set whose elements are the indices of the first $t$-number of smallest
elements in a.

Proof: As shown in Fig. 7, without loss of generality $\mathbf{d}_v$ is assumed to be in the plane formed by $\mathbf{a}$ and $\mathbf{b}$ as the perpendicular component of $\mathbf{d}_v$ to the plane does not have any effect on the value of $q(\mathbf{d}_v)$. There are two geometrical cases for $\theta_2$, i.e., $\theta_2 = \pi + \theta - \theta_1$ for $\theta + \theta_1 + \theta_2 > \pi$ or $\theta_2 = \pi - \theta - \theta_1$ otherwise. By the definition, $\theta_1$ is given by

$$\theta_1 = \cos^{-1}\left(\frac{\mathbf{a}^T \mathbf{d}_v}{\|\mathbf{a}\|} \right),$$

and $\cos(\theta_1)$ is a monotonically decreasing function for $\theta_1 \in [0, \pi]$. Hence, for a fixed $t$, i.e., the number of 1’s in $\mathbf{d}_v$, the minimum or the maximum of $\theta_1$ occurs at the summation of the maximum or the minimum $t$-number of elements in $\mathbf{a}$.

**QUARTIC EQUATION**

**Proposition A.2:** $q(\mathbf{d}_v)$ in (3) is equivalent to

$$q(\theta_1) = a\sqrt{t}x - bt(x \cos \theta \pm \sqrt{1 - x^2} \sin \theta)^2,$$

where $x = \cos \theta_1$, and the following inequality is satisfied if $\theta_1$ takes any values between $\theta_1^\ast$ and $\theta_1$:

$$\min q(\theta_1) \leq \min q(\mathbf{d}_v).$$

Proof: The magnitude of $\mathbf{d}_v$ is $\sqrt{t}$ and (3) becomes

$$q(\mathbf{d}_v) = a\sqrt{t} \cos \theta_1 - bt \cos^2 \theta_2.$$

Substitute $\theta_2 = \pi \pm \theta - \theta_1$ into the above

$$q(\theta_1) = a\sqrt{t} \cos \theta_1 - bt \cos^2 (\pm - \theta_1),$$

where $x = \cos \theta_1$, and the following inequality is satisfied if $\theta_1$ takes any values between $\theta_1^\ast$ and $\theta_1$:

$$\min q(\theta_1) \leq \min q(\mathbf{d}_v).$$

**Proposition A.3:** Let $q(\theta_1^\ast) = \min q(\theta_1)$ and $\theta_1^\ast$ is equal to $\theta_1^\ast$, $\theta_1$ or $\cos^{-1} x^\ast$, where $x^\ast$ is the solution of quartic polynomial equation: $\sum_{i=0}^{4} w_i x_i = 0$, whose coefficients are given by the following two cases:

$$w_4 = 4b^2 t^2 \left[4 \sin^2 \theta \cos^2 \theta + (2 \cos^2 \theta - 1)^2 \right],$$

$$w_3 = -4abt \sqrt{t} (2 \cos^2 \theta - 1),$$

$$w_2 = -16b^2 t^2 \sin^2 \theta \cos^2 \theta + a^2 t^2 - 4b^2 t^2 (2 \cos^2 \theta - 1)^2,$$

$$w_1 = 4abt \sqrt{t} (2 \cos^2 \theta - 1),$$

$$w_0 = 4b^2 t^2 \sin^2 \theta \cos^2 \theta - a^2 t,$$

or

$$w_4 = 4b^2 t^2 (2 \cos^2 \theta - 1)^2,$$

$$w_3 = -4abt \sqrt{t} (2 \cos^2 \theta - 1),$$

$$w_2 = a^2 t^2 - 4b^2 t^2 (2 \cos^2 \theta - 1)^2,$$

$$w_1 = 4abt \sqrt{t} (2 \cos^2 \theta - 1),$$

$$w_0 = 4b^2 t^2 \sin^2 \theta \cos^2 \theta - a^2 t,$$

and $x \in [-1, 1]$.

Proof: $\theta_1^\ast$ will occur either on the boundary, i.e., $\theta_1^\ast$ or $\tilde{\theta}_1^\ast$, or the angles in $(\tilde{\theta}_1^\ast, \tilde{\theta}_1^\ast)$, where the derivative of $\frac{d}{dx} q(\theta_1^\ast)$ is equal to zero.

$$\frac{d}{dx} q(\theta_1^\ast) = \frac{d}{dx} q(\theta_1^\ast) \ dx \ \frac{d}{dx} \theta_1^\ast = -\frac{d}{dx} q(\theta_1^\ast) \sin \theta_1^\ast = 0.$$

Immediate solutions from $\sin \theta_1^\ast = 0$ are $\theta_1^\ast = 0$ or $\pi$ and they would be either on the boundary of the domain of $\theta_1$ or outside of the boundary. Hence, they are automatically considered when the boundary values are checked. The remaining $\theta_1^\ast$ values to be checked are the ones making the derivative equal to zero. Take the derivative

$$\frac{d}{dx} q(\theta) = a\sqrt{t} - 2bt (2 \cos^2 \theta - 1) x$$

$$\mp 2bt \sin \theta \cos \sqrt{1 - x^2} + 2bt \sin \theta \cos \theta \frac{\cos \frac{x^2}{\sqrt{1 - x^2}}}{\sqrt{1 - x^2}} = 0.$$ 

After squaring both sides and some algebraic manipulations, which is tedious and omitted, it leads to the two quartic polynomials in $x$.

**INEQUALITY FOR THE UPPER BOUND**

**Proposition A.4:** The minimum of $q(\mathbf{d}_v)$ is bounded above by

$$\min_{\mathbf{d}_v \in \mathbb{D}_v} q(\mathbf{d}_v) \leq q(\mathbf{d}_v),$$

where

$$q(\mathbf{d}_v) = [\alpha \mathbf{a}^T \mathbf{A}_v \mathbf{d}_v + p(\mathbf{d}_v)] \times (1 + \alpha)^{-1},$$

$$p(\mathbf{d}_v) := (\mathbf{a}_1^T - \hat{\mathbf{a}}_1^T) \mathbf{A}_v \mathbf{d}_v - \mathbf{d}_v^T \mathbf{b} \mathbf{b}^T \mathbf{d}_v,$$

$$\mathbf{a}_1^T := \mathbf{s}^T (I_n \otimes \mathbf{s}^T) L,$$

$$\hat{\mathbf{a}}_1^T := \mathbf{s}^T \mathbf{k} \mathbf{s}^T (I_n \otimes \mathbf{I}^T) L \times m^{-1},$$

$$\mathbf{b} := \mathbf{A}_v^T L (I + \mathbf{I}^T)^T \mathbf{s} \times (\sqrt{2m})^{-1},$$

$$\mathbf{d}_v := \arg \min_{\mathbf{d}_v \in \mathbb{D}_v} p(\mathbf{d}_v),$$

$$\alpha := \mathbf{I}^T \mathbf{A}_v \mathbf{d}_v.$$

Proof: Recall (5) in Appendix and rearrange it as follows:

Minimize $q(\mathbf{d}_v) = \left[\alpha \mathbf{a}_1^T - \hat{\mathbf{a}}_1^T \right] \mathbf{A}_v \mathbf{d}_v - \mathbf{d}_v^T \mathbf{b} \mathbf{b}^T \mathbf{d}_v$

$$= \frac{1}{1 + \alpha} \left\{[(1 + \alpha) \mathbf{a}_1^T - \hat{\mathbf{a}}_1^T] \mathbf{A}_v \mathbf{d}_v - \mathbf{d}_v^T \mathbf{b} \mathbf{b}^T \mathbf{d}_v \right\},$$

min $p(\mathbf{d}_v)$ is the minimizing solution of only parts of $q(\mathbf{d}_v)$ and the corresponding solution, $(\mathbf{d}_v, \hat{\alpha})$, is substituted into $q(\mathbf{d}_v)$, which is equal to $q(\mathbf{d}_v)$. Hence, min $q(\mathbf{d}_v) \leq q(\mathbf{d}_v)$.

**Proposition A.5:** The following inequality is satisfied:

$$q(\mathbf{d}_v) \leq q(\mathbf{d}_v),$$

where

$$\mathbf{d}_v = T \left[ \arg \min_{\mathbf{d}_v \in \mathbb{D}_v} p(\mathbf{d}_v) \right].$$
i.e., $T(\cdot)$ is the operator to transform $d_v$ in $D_{ev}$ to the corresponding $d_v$ in $D_v$. For example, for $l = 3$, $d_v = [1 0 0]$, then $d_v = T(d_v) = T([1 0 0]) = [1 1 0]^T$.

Proof) As $d_v$ is transformed from the minimizing solution of $\hat{p}(d_v)$ by $T(\cdot)$. By the definitions, $p(d_v)$ is greater than or equal to $\min p(d_v)$. Hence, $q(d_v)$ is also greater than or equal to $q(d_v)$. \[\blacksquare\]

**Proof of Theorem 2.5**

As each submodule is part of a whole network, the modularity definition for a submodule is as follows [2]:

Maximize $Q(s, A^{sg}) = \frac{1}{4m} s^T B^{sg} s$, where

$$B^{sg} = A^{sg} - \frac{1}{2m} k^{sg} k^{sgT} - \text{diag} \{\tilde{k}^{sg}, \tilde{k}^{sg}, \ldots, \tilde{k}^{sg}\}$$

$$+ \frac{1}{2m} \text{diag} \{k^{sg}_1 1^T k^{sg}, k^{sg}_2 1^T k^{sg}, \ldots, k^{sg}_{n_g} 1^T k^{sg}\},$$

$B^{sg}$ is scaled by the last two terms in order to evaluate the modularity in the whole network, $A^{sg}$ is the adjacency matrix including only the concerned submodule,

$$\tilde{k}^{sg}_i = \sum_{j=1}^{n_g} A^{sg}_{ij},$$

for $i = 1, 2, \ldots, n_g$,

$$k^{sg} = \begin{bmatrix} \sum_{j=1}^{n} A_{i,j} & \sum_{j=1}^{n} A_{i,j} & \ldots & \sum_{j=1}^{n} A_{i,j} \end{bmatrix}^T,$$

$\{l_1, l_2, \ldots, l_{n_g}\}$ are the indices including the nodes that belong to the submodule, and $n_g$ is the number of nodes in the submodule. Re-arrange $Q$ for submodule

$$Q(s, A^{sg}) = s^T \frac{1}{4m} \left(A^{sg} - \frac{1}{2m} k^{sg} k^{sgT}\right) s$$

$$- \frac{1}{4m} s^T \begin{bmatrix} s_1 \tilde{k}^{sg} \\ s_2 \tilde{k}^{sg} \\ \vdots \\ s_{n_g} \tilde{k}^{sg} \end{bmatrix}^T + \frac{1}{8m^2} s^T \text{diag} \begin{bmatrix} k^{sg}_1 1^T k^{sg} \\ k^{sg}_2 1^T k^{sg} \\ \vdots \\ k^{sg}_{n_g} 1^T k^{sg} \end{bmatrix} s$$

$$= s^T \frac{1}{4m} \left(A^{sg} - \frac{1}{2m} k^{sg} k^{sgT}\right) s - \frac{s^T \text{diag}[s]}{4m} \tilde{k}^{sg},$$

where $\tilde{k}^{sg}$ is the vector constructed by $\tilde{k}^{sg}_i$. Note that perturbations only occur in the submodule, i.e. $A^{sg} = A^{sg} + \Delta_{11}$, hence

$$k^{sg}_i = k^{sg}_i + \delta_k$$

$$\tilde{k}^{sg}_i = \tilde{k}^{sg}_i + \delta_k.$$

Then,

$$Q(\Delta_{11}) = s^T \frac{1}{4m} \left(A^{sg} - \frac{1}{2m} k^{sg} k^{sgT}\right) s^*$$

$$- \frac{s^T \text{diag}[s]}{4m} \tilde{k}^{sg}_i + \frac{s^T \text{diag}[s]}{4m} \tilde{k}^{sg}_i \frac{(s^* 1^T)^T}{8m^2} k^{sg}_i,$$

where $s^* = \text{argmax} Q(s, A^{sg})$. The worst-case analysis problem is given by

$$\text{Minimize } Q(\Delta_{11}) = s^* T \frac{1}{4m} \left(A^{sg} - \frac{1}{2m} k^{sg} k^{sgT}\right) s^*$$

$$- \frac{s^T \text{diag}[s]}{4m} \tilde{k}^{sg}_i + \frac{s^T \text{diag}[s]}{4m} \tilde{k}^{sg}_i \frac{(s^* 1^T)^T}{8m^2} k^{sg},$$

where the first term in the right hand side has exactly the same form as the one in the whole network and $m_g$ can be written as

$$2m_g = 1^T A_g 1 = 1^T A 1 + 1^T \Delta_{11} 1 = 2m(1 + \alpha^{sg}),$$

and $\alpha^{sg} = \delta^{sg}_m/m$. From the same logic as before, there are two extreme perturbations and

$$-\frac{m_g}{m} \leq \alpha^{sg} \leq \frac{n_g(n_g - 1)}{2m} - \frac{m_g}{m}.$$