PAPER

Network Resonance Method: Estimating Network Structure from the Resonance of Oscillation Dynamics

Satoshi FURUTANI[†*a)], **Chisa TAKANO**[††b)], ***and*** **Masaki AIDA**[†c)], ***Members***

SUMMARY Spectral graph theory, based on the adjacency matrix or the Laplacian matrix that represents the network topology and link weights, provides a useful approach for analyzing network structure. However, in large scale and complex social networks, since it is difficult to completely know the network topology and link weights, we cannot determine the components of these matrices directly. To solve this problem, we propose a method for indirectly determining the Laplacian matrix by estimating its eigenvalues and eigenvectors using the resonance of oscillation dynamics on networks.

key words: *Laplacian matrix, spectral graph theory, resonance*

1. Introduction

There are many network systems in the real world, such as the Internet, the World Wide Web (WWW), and social networks. Accordingly, networks are studied in many research fields, such as mathematics, physics, engineering, biology and economics.

Spectral graph theory [1] gives a useful approach for analyzing network structure based on the adjacency matrix or the Laplacian matrix. In particular, the eigenvalues of the Laplacian matrix give important information for describing the dynamic properties of the network; for example, mixing speed of Markov chains [2], [3], synchronization on networks [4]–[7] and virus propagation in networks [8], [9].

If we want to use a matrix to represent a network structure, we have to know all matrix components, i.e. the presence and strength of links. However, in large scale and complex social networks, it is difficult to know the values of all matrix components. This is because it is difficult to observe the presence of links and their strength over the whole network. Typical examples are social networks formed by human relationships. In these networks, the strength of connections between people, i.e. friendship, is likely to be inhomogeneous and can never be quantitatively measured. The unavailability of link strength values prevents the determination of network structure even if we know the complete topological structure of the network.

Given the above background, we propose a method for estimating the structure of a network by observing the network's reaction to external perturbations. The key to our method is spectral decomposition of the Laplacian matrices. According to the spectral decomposition, Laplacian matrices which describe the structure of undirected graphs can be uniquely described by its eigenvalues and eigenvectors. This means the eigenvalues and eigenvectors have the information about the structure of networks. Thus, if we can get all the eigenvalues and eigenvectors of a Laplacian matrix without *a priori* information about the network structure, we can reproduce the original Laplacian matrix, i.e. we can determine the network structure indirectly. Moreover, the above concept can be applied to some particular directed graphs, since the (asymmetric) Laplacian matrix of directed graphs which satisfy the particular condition can be symmetrized [10], [11].

In this paper, we explain our method for estimating the structure of networks, called the *network resonance method*. This method estimates the structure of a network by estimating the eigenvalues [12] and eigenvectors of a Laplacian matrix by observing the oscillation dynamics on the network. Here the eigenvectors can be estimated by combining the techniques for estimating the absolute value of their components [13] and determining their sign.

This paper is organized as follows. Section 2 explains the Laplacian matrix and oscillation model on networks as an introduction to the description of our method. Section 3 proposes the network resonance method. Section 4 shows its effectiveness through numerical experiments. Section 5 discusses the issues of our method and introduces related work. Finally, we conclude this paper in Sect. 6.

2. Preliminary

2.1 Laplacian Matrix

Let $G = (V, E)$ be an undirected graph, where $V = \{1, 2, \ldots, n\}$ is the set of nodes and E ($\subset V \times V$) is the set of links in G. The (weighted) adjacency matrix $\boldsymbol{A} = \left(A_{ij}\right)$ is the $n \times n$ matrix defined as

$$A_{ij} := \begin{cases} k_{ij} & ((i,j) \in E), \\ 0 & ((i,j) \notin E). \end{cases} \tag{1}$$

In undirected graph, $\boldsymbol{A}$ is a real symmetric matrix, because $k_{ij} = k_{ji}$. The degree of each node is $d_i := \sum_{j=1}^{n} k_{ij}$. The

Manuscript received June 3, 2018.
Manuscript revised September 8, 2018.
Manuscript publicized October 18, 2018.

[†]The authors are with the Graduate School of Systems Design, Tokyo Metropolitan University, Hino-shi, 191-0065 Japan.
[††]The author is with the Graduate School of Information Sciences, Hiroshima City University, Hiroshima-shi, 731-3194 Japan.
[*]Presently, with the NTT Secure Platform Laboratories, NTT Corporation.
a) E-mail: furutani.satoshi@lab.ntt.co.jp
b) E-mail: takano@hiroshima-cu.ac.jp
c) E-mail: aida@tmu.ac.jp

DOI: 10.1587/transcom.2018EBP3160

degree matrix is defined as

$$\boldsymbol{D} := \mathrm{diag}(d_1, d_2, \ldots, d_n),$$

and the Laplacian matrix is defined as follows [14]:

$$\boldsymbol{L} := \boldsymbol{D} - \boldsymbol{A}. \tag{2}$$

In undirected graph, $\boldsymbol{L}$ is also a real symmetric matrix, the same as $\boldsymbol{A}$. The Laplacian matrix is also called graph Laplacian and is often used to describe the diffusion phenomena on networks. Additionally, normalized Laplacian matrix $\boldsymbol{N} := \boldsymbol{D}^{-1/2}\boldsymbol{L}\boldsymbol{D}^{-1/2}$, that is a Laplacian matrix scaled by node degree, has been used for random walk analysis of networks [15].

Next, we define the Laplacian matrix of directed graph $\mathcal{G} = (V, \mathcal{E})$, where $\mathcal{E}$ is the set of directed links. The (weighted) adjacency matrix $\boldsymbol{\mathcal{A}} = \left(\mathcal{A}_{ij}\right)$ of a directed graph $\mathcal{G}$ is defined as

$$\mathcal{A}_{ij} := \begin{cases} w_{ij} & ((i \to j) \in \mathcal{E}), \\ 0 & ((i \to j) \notin \mathcal{E}). \end{cases} \tag{3}$$

The out-degree of each node is $d_i^{\mathrm{out}} := \sum_{j=1}^{n} w_{ij}$. The degree matrix is defined as $\boldsymbol{\mathcal{D}} := \mathrm{diag}(d_1^{\mathrm{out}}, d_2^{\mathrm{out}}, \ldots, d_n^{\mathrm{out}})$, and the Laplacian matrix $\boldsymbol{\mathcal{L}}$ of a directed graph is defined as

$$\boldsymbol{\mathcal{L}} := \boldsymbol{\mathcal{D}} - \boldsymbol{\mathcal{A}}. \tag{4}$$

In general, $\boldsymbol{\mathcal{L}}$ is asymmetrical because $w_{ij} \neq w_{ji}$. In this paper, we call matrix $\boldsymbol{\mathcal{L}}$ an asymmetric Laplacian matrix.

If we assume that the left eigenvector ${}^t\boldsymbol{m}$ of $\boldsymbol{\mathcal{L}}$, where ${}^t\boldsymbol{m} = (m_1, m_2, \ldots, m_n)$, that is associated with a zero eigenvalue satisfies

$${}^t\boldsymbol{m}\,\boldsymbol{\mathcal{L}} = (0, 0, \ldots, 0), \tag{5}$$

and $m_i > 0$ for all i satisfy

$$m_i\, w_{ij} = m_j\, w_{ji} \quad (\equiv k_{ij}), \tag{6}$$

then the link asymmetry of $\boldsymbol{\mathcal{L}}$ can be expressed by using symmetric Laplacian matrix $\boldsymbol{L}$. Namely, $\boldsymbol{\mathcal{L}}$ can be expressed as follows.

$$\boldsymbol{\mathcal{L}} = \boldsymbol{M}^{-1}\boldsymbol{L}, \tag{7}$$

where $\boldsymbol{M} := \mathrm{diag}(m_1, m_2, \ldots, m_n)$. This implies that some particular asymmetry of links can be reduced to node characteristics like Fig. 1; for more detail, see [10], [11].

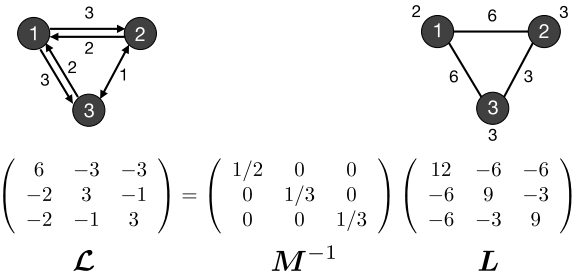

Fig. 1 An example of reducing link asymmetry to node characteristics.

2.2 Scaled Laplacian Matrix

As mentioned in the previous subsection, normalized Laplacian matrix $\boldsymbol{N}$ is the Laplacian matrix that is scaled by node degree d_i. To generalize this, we introduce scaled Laplacian matrix $\boldsymbol{S}$ that is the Laplacian matrix scaled by arbitrary node characteristic m_i (> 0). Scaled Laplacian matrix $\boldsymbol{S}$ is defined as follows:

$$\boldsymbol{S} := \boldsymbol{M}^{-1/2}\,\boldsymbol{L}\,\boldsymbol{M}^{-1/2}. \tag{8}$$

Since scaled Laplacian matrix $\boldsymbol{S}$ is a nonnegative-definite matrix and its minimum eigenvalue is 0, we sort its eigenvalues in ascending order as

$$0 = \lambda_0 \leq \lambda_1 \leq \cdots \leq \lambda_{n-1},$$

and we define the eigenvector associated with eigenvalue λ_μ as $\boldsymbol{v}_\mu$ ($\mu = 0, 1, \ldots, n-1$), that is

$$\boldsymbol{S}\,\boldsymbol{v}_\mu = \lambda_\mu\,\boldsymbol{v}_\mu.$$

We can choose the eigenvectors as the eigenbasis, that is, they are mutually orthogonal with length of 1, as

$$\boldsymbol{v}_\mu \cdot \boldsymbol{v}_\nu = \delta_{\mu\nu},$$

where $\delta_{\mu\nu}$ is the Kronecker delta.

Since scaled Laplacian matrix $\boldsymbol{S}$ is a real symmetric matrix, $\boldsymbol{S}$ can be diagonalized by using orthogonal matrix $\boldsymbol{V}$ as

$$\boldsymbol{\Lambda} = {}^t\boldsymbol{V}\,\boldsymbol{S}\,\boldsymbol{V}, \tag{9}$$

where $\boldsymbol{\Lambda} := \mathrm{diag}(\lambda_0, \lambda_1, \ldots, \lambda_{n-1})$ is the diagonal matrix of the eigenvalues, and $\boldsymbol{V} := (\boldsymbol{v}_0, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_{n-1})$ is the orthogonal matrix made by the eigenbasis. Equivalently, we have

$$\boldsymbol{S} = \boldsymbol{V}\,\boldsymbol{\Lambda}\,{}^t\boldsymbol{V}. \tag{10}$$

This means that scaled Laplacian matrix $\boldsymbol{S}$ can be completely determined if we have all the information about the eigenvalues and eigenvectors of $\boldsymbol{S}$.

Finally, we show the relationship between asymmetric Laplacian matrix $\boldsymbol{\mathcal{L}}$ for directed networks and the corresponding scaled Laplacian matrix $\boldsymbol{S}$. By multiplying $\boldsymbol{M}^{-1/2}$ by the characteristic equation from the left, we obtain

$$\begin{aligned} &\boldsymbol{M}^{-1/2}\boldsymbol{S}\,\boldsymbol{v}_\mu = \lambda_\mu\,\boldsymbol{M}^{-1/2}\,\boldsymbol{v}_\mu \\ &\Leftrightarrow \boldsymbol{M}^{-1}\boldsymbol{L}\,(\boldsymbol{M}^{-1/2}\,\boldsymbol{v}_\mu) = \lambda_\mu\,(\boldsymbol{M}^{-1/2}\,\boldsymbol{v}_\mu) \\ &\Leftrightarrow \boldsymbol{\mathcal{L}}\,\boldsymbol{u}_\mu = \lambda_\mu\,\boldsymbol{u}_\mu, \end{aligned} \tag{11}$$

with

$$\boldsymbol{u}_\mu := \boldsymbol{M}^{-1/2}\,\boldsymbol{v}_\mu. \tag{12}$$

Thus, we recognize scaled Laplacian matrix $\boldsymbol{S}$ has the same eigenvalue as $\boldsymbol{\mathcal{L}}$, and their eigenvectors are related as given

by (12).

This means that the directed graph described by asymmetric Laplacian matrix $\boldsymbol{\mathcal{L}}$ can be analyzed via the corresponding (symmetric) scaled Laplacian matrix $\boldsymbol{S}$. More concretely, the structural and dynamical properties of the directed graph described by $\boldsymbol{\mathcal{L}}$ can be analyzed via the eigenvalues and eigenvectors of $\boldsymbol{S}$. In general, it is difficult to analyze asymmetric matrices because they are not always diagonalizable. Hence this fact is fruitful for analyzing directed graphs that satisfy (6).

2.3 Oscillation Model on Networks

To describe the propagation of activity on networks, Aida et al. proposed an oscillation model on networks in [10]. Let us assume a system in which each node has a state value, and its state influences the states of adjacent nodes. Let x_i of node i be its displacement from equilibrium, and let the restoring force be proportional to the difference of the displacements of the adjacent nodes.

Note that this is a kind of the most simple and universal model. If all nodes have the same value of the state, it is natural to assume no influence occurs between nodes. In addition, it is natural to assume the strength of influence between nodes are given by increasing function $f(\Delta x)$ of the difference, Δx, of the values of node state. Even if function $f(\Delta x)$ is nonlinear, we can obtain a linear relation for small values of Δx by applying the first order approximation of the Taylor expansion. Namely,

$$f(\Delta x) = k_{ij}\Delta x + O\left((\Delta x)^2\right),$$

where k_{ij} is a positive constant corresponding to the spring constant. Thus, the oscillation model is an universal model suitable for many different influence models.

The oscillation model sets the spring constant between nodes i and j to k_{ij} and the mass of node i to m_i. Here, we consider the situation that we impose a forced oscillation with frequency ω on node j as an external force, and suppose that the damped force is proportional to the velocity $\mathrm{d}x_i(t)/\mathrm{d}t$ of node i. The equation of motion of the forced oscillation on networks can be written by using the Laplacian matrix as follows:

$$\boldsymbol{M}\frac{\mathrm{d}^2\boldsymbol{x}(\omega,t)}{\mathrm{d}t^2} + \boldsymbol{M}\gamma\frac{\mathrm{d}\boldsymbol{x}(\omega,t)}{\mathrm{d}t} + \boldsymbol{L}\,\boldsymbol{x}(\omega,t) = (F\cos\omega t)\,\mathbf{1}_{\{j\}} \tag{13}$$

or

$$\frac{\mathrm{d}^2\boldsymbol{x}(\omega,t)}{\mathrm{d}t^2} + \gamma\frac{\mathrm{d}\boldsymbol{x}(\omega,t)}{\mathrm{d}t} + \boldsymbol{\mathcal{L}}\,\boldsymbol{x}(\omega,t) = \frac{F\cos\omega t}{m_j}\,\mathbf{1}_{\{j\}}, \tag{14}$$

where γ and F are constants, and $\mathbf{1}_{\{j\}}$ is the n-dimensional vector whose j-th component is 1; all other components are 0. Then,

$$\boldsymbol{x}(\omega,t) = {}^t(x_1(\omega,t),\, x_2(\omega,t),\, \ldots,\, x_n(\omega,t)).$$

By using vector $\boldsymbol{y}(\omega,t) = \boldsymbol{M}^{1/2}\boldsymbol{x}(\omega,t)$, (13) and (14) can be rewritten as

$$\frac{\mathrm{d}^2\boldsymbol{y}(\omega,t)}{\mathrm{d}t^2} + \gamma\frac{\mathrm{d}\boldsymbol{y}(\omega,t)}{\mathrm{d}t} + \boldsymbol{S}\boldsymbol{y}(\omega,t) = \frac{F\cos\omega t}{\sqrt{m_j}}\,\mathbf{1}_{\{j\}}. \tag{15}$$

The stationary solution of (13) can be written by using eigenvalues λ_μ and associated eigenvectors $\boldsymbol{v}_\mu$ of $\boldsymbol{S}$ as

$$\boldsymbol{x}(\omega,t) = \boldsymbol{M}^{-1/2}\sum_{\mu=0}^{n-1} A_\mu(\omega)\cos\left(\omega t + \theta_\mu(\omega)\right)\boldsymbol{v}_\mu, \tag{16}$$

where $A_\mu(\omega)$ and $\theta_\mu(\omega)$ are the amplitude and the phase for eigenmode μ, respectively. They are expressed as

$$A_\mu(\omega) = \frac{F\,v_\mu(j)}{\sqrt{m_j}}\frac{1}{\sqrt{(\omega_\mu^2-\omega^2)^2+(\gamma\,\omega)^2}}, \tag{17}$$

$$\theta_\mu(\omega) = \arctan\left(\frac{\gamma\,\omega}{\omega^2-\omega_\mu^2}\right). \tag{18}$$

Note that, eigenfrequency $\omega_\mu = \sqrt{\lambda_\mu}$.

3. Network Resonance Method

In this section, we explain the network resonance method that estimates the structure of networks without *a priori* information about the links of networks. In the framework of the network resonance method, we estimate network structure from the network's reaction to external perturbations. Here, the external perturbation corresponds to a periodic external force, $F\cos\omega t$, of (13) and the network's reaction corresponds to the amplitude $a_i(\omega) := \max_t |x_i(\omega,t)|$ observed at each node i. Note that node centrality (the strength of node activity) can be understood as the oscillation energy that is proportional to the square of the amplitude [16], [17]. Hence it is expected that oscillation energy and amplitude of nodes are observable state quantities in the real networks.

Since node amplitude $a_i(\omega)$ increases sharply around each eigenfrequency because of resonance, we obtain a curve with several peaks as the node reaction. The position and height of each peak correspond to the eigenvalues and eigenvectors, respectively. Thus, it is expected that we can estimate the eigenvalues and eigenvectors by focusing on these peaks. Figure 2 shows a schematic diagram of the network resonance method.

The method for estimating eigenvalues and eigenvectors is detailed below.

3.1 Method for Estimating Eigenvalues

This subsection explains the method that estimates the eigenvalues of a scaled Laplacian matrix from the amplitudes of oscillation dynamics without *a priori* information about the network structure.

Node amplitude $a_i(\omega)$ includes contributions from all

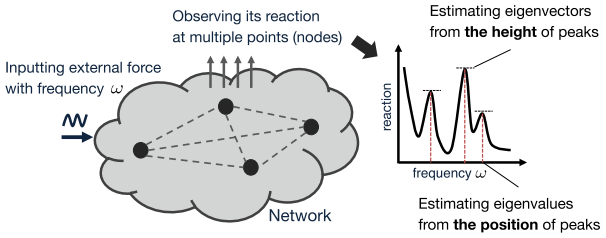

Fig. 2 Network resonance method.

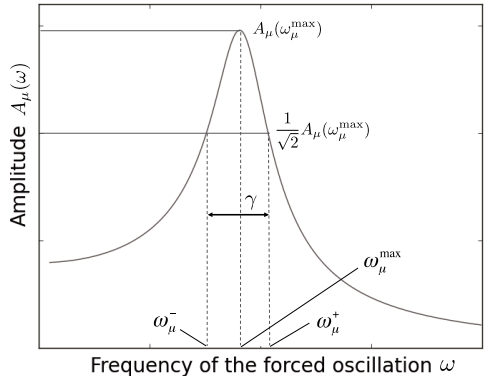

Fig. 3 Determining frequencies ω_μ^- and ω_μ^+ from amplitude $A_\mu(\omega)$.

eigenmodes μ ($\mu = 0, 1, \ldots, n-1$). Here, let us consider amplitude $A_\mu(\omega)$ for a single eigenmode μ. From (17), amplitude $A_\mu(\omega)$ increases sharply around $\omega \simeq \omega_\mu$. This phenomenon is called *resonance*. Amplitude $A_\mu(\omega)$ takes maximal value at

$$\omega = \sqrt{\omega_\mu^2 - \frac{\gamma^2}{2}} \quad (=: \omega_\mu^{\max}). \tag{19}$$

We then define ω_μ^- and ω_μ^+ ($\omega_\mu^- < \omega_\mu^+$) as the frequencies that satisfy

$$A_\mu(\omega_\mu^\pm) = \frac{1}{\sqrt{2}} A_\mu(\omega_\mu^{\max}), \tag{20}$$

like Fig. 3.

If $\gamma \ll \omega_\mu$, the eigenfrequency ω_μ and the damping factor γ can be approximately represented as

$$\omega_\mu \simeq 2\,\omega_\mu^\pm - \sqrt{2(\omega_\mu^\pm)^2 - (\omega_\mu^{\max})^2}, \tag{21}$$

$$\gamma \simeq \omega_\mu^+ - \omega_\mu^-, \tag{22}$$

where the double-sign of (21) corresponds. $\omega_\mu^-, \omega_\mu^+$ and $\omega_\mu^{\max}$ are the values obtained by observing amplitude $A_\mu(\omega)$, so we can estimate the eigenvalue λ_μ ($= \omega_\mu^2$) by substituting them into (21). In the same way, we can estimate the value of damping factor γ.

Although we cannot actually observe amplitude $A_\mu(\omega)$ for each μ alone, it is assumed that the contribution of the amplitude $A_\mu(\omega)$ is dominant relative to the amplitudes $a_i(\omega)$ around $\omega \simeq \omega_\mu$. This yields the expectation that we can estimate the eigenvalues of $\boldsymbol{S}$ by obtaining the values corresponding to ω_μ^-, ω_μ^+ and $\omega_\mu^{\max}$ from the peaks of amplitude $a_i(\omega)$. In fact, it is reported that this method makes it possible to estimate the eigenvalues of $\boldsymbol{S}$ with high accuracy [12].

3.2 Method for Estimating Eigenvectors

In this section, we explain our method for estimating the eigenvectors of a scaled Laplacian matrix. In estimating eigenvectors, we focus on not the position but the height of amplitude peaks.

Let us decompose node displacement $x_i(\omega, t)$ into eigenmode $x_{i,\mu}(\omega, t)$ as $x_i(\omega, t) := \sum_{\mu=0}^{n-1} x_{i,\mu}(\omega, t)$. The amplitude $c_{i,\mu}(\omega) = \max_t |x_{i,\mu}(\omega, t)|$ of the eigenmode is expressed as follows:

$$c_{i,\mu}(\omega) = \left| \frac{1}{\sqrt{m_i\, m_j}} \frac{F\, v_\mu(i)\, v_\mu(j)}{\sqrt{(\omega_\mu^2 - \omega^2)^2 + (\gamma\,\omega)^2}} \right|. \tag{23}$$

By substituting $\omega = \omega_\mu$ into (23) and rearranging the terms, we get

$$|v_\mu(i)\, v_\mu(j)| = \frac{\sqrt{m_i\, m_j}\, \gamma\, \omega_\mu\, c_{i,\mu}(\omega_\mu)}{F}. \tag{24}$$

When $i = j$, (24) becomes

$$|v_\mu(i)| = \sqrt{\frac{m_i\, \gamma\, \omega_\mu\, c_{i,\mu}(\omega_\mu)}{F}}. \tag{25}$$

In (24) and (25), F is the known parameter that represents the strength of the external force and γ and ω_μ can be estimated from (21) and (22). Thus, we can obtain the absolute values of the i and j-th components, $v_\mu(i)$ and $v_\mu(j)$, of eigenvector $\boldsymbol{v}_\mu$ if we can estimate the unknown values of m_i, m_j and $c_{i,\mu}(\omega_\mu)$. We explain the technique used to estimate m_i, m_j and $c_{i,\mu}(\omega_\mu)$ below.

First, we show how to estimate $c_{i,\mu}(\omega_\mu)$. Let us divide node displacement $x_i(\omega, t)$ into the term for eigenmode μ and all other terms as follows.

$$x_i(\omega, t) = c_{i,\mu}(\omega)\cos(\omega\, t + \theta_\mu(\omega)) + \sum_{\substack{\nu=0 \\ \nu\neq\mu}}^{n-1} c_{i,\nu}(\omega)\cos(\omega\, t + \theta_\nu(\omega)). \tag{26}$$

By substituting $\omega = \omega_\mu$ into (26), and using $\theta_\mu(\omega_\mu) \simeq \pi/2$ and $\theta_\nu(\omega_\mu) \simeq 0$ obtained from (18), (26) can be rewritten as

$$x_i(\omega_\mu, t) = -c_{i,\mu}(\omega_\mu)\sin(\omega_\mu\, t) + \sum_{\substack{\nu=0 \\ \nu\neq\mu}}^{n-1} c_{i,\nu}(\omega_\mu)\cos(\omega_\mu\, t) = \sqrt{\alpha_\mu^2 + \beta_\mu^2}\,\sin(\phi_\mu - \omega_\mu\, t), \tag{27}$$

with

$$\alpha_\mu := c_{i,\mu}(\omega_\mu),\ \beta_\mu := \sum_{\substack{\nu=0 \\ \nu\neq\mu}}^{n-1} c_{i,\nu}(\omega_\mu),$$

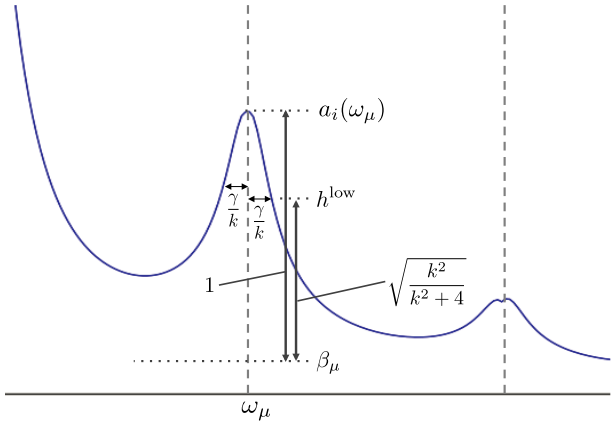

Fig. 4 Determining the contribution, β_μ, of the eigenmodes other than the eigenmode of interest.

$$\cos\phi_\mu := \frac{\alpha_\mu}{\sqrt{\alpha_\mu^2+\beta_\mu^2}}.$$

Because $a_i(\omega_\mu) := \max_t |x_i(\omega_\mu, t)| = \sqrt{\alpha_\mu^2+\beta_\mu^2}$, we obtain

$$c_{i,\mu}(\omega_\mu) = \sqrt{a_i(\omega_\mu)^2 - \beta_\mu^2}. \tag{28}$$

Thus, if we have the value of node amplitude $a_i(\omega_\mu)$ and the contribution $\beta_\mu = \sum_{\nu=0,\,\nu\neq\mu}^{n-1} c_{i,\nu}(\omega_\mu)$ from the eigenmodes other than eigenmode μ, we can obtain the value of $c_{i,\mu}(\omega_\mu)$.

To determine β_μ, we focus on (17). By substituting $\omega = \omega_\mu \pm \gamma/k$ into $A_\mu(\omega)$, we obtain

$$A_\mu\left(\omega_\mu \pm \gamma/k\right) = \sqrt{\frac{k^2}{k^2+4}}\,\frac{F\,v_\mu(j)}{\sqrt{m_j}\,\omega_\mu\,\gamma} = \sqrt{\frac{k^2}{k^2+4}}\,A_\mu(\omega_\mu). \tag{29}$$

Therefore, let us define h^{low} and h^{high} ($h^{\text{low}} < h^{\text{high}}$) as heights $a_i(\omega_\mu + \gamma/k)$ and $a_i(\omega_\mu - \gamma/k)$. Then, let β_μ be the external dividing point whose external ratio of $a_i(\omega_\mu)$ to h^{low} is $1 : \sqrt{k^2/(k^2+4)}$ (Fig. 4). Finally, by substituting β_μ and $a_i(\omega_\mu)$ into (28), we obtain amplitude $c_{i,\mu}(\omega_\mu)$ of eigenmode μ.

This method always succeeds if the contribution, β_μ, from the eigenmodes other than eigenmode μ around the peak of eigenmode μ can be regarded as constant. While in practice contribution β_μ around the peak is not constant, but rather is proportional to ω^{-2}, we can consider that contribution β_μ is virtually constant because we focus on the narrow region of γ/k. Furthermore, the reason for choosing h^{low} over h^{high} is that h^{high} makes a larger contribution to β_μ than h^{low}, and thus would degrade the precision of the estimated values.

Next, we explain how to estimate node mass m_i. For that, we focus on the oscillation energy. Since $x_i(\omega, t)$ is periodic oscillation with frequency ω in the steady state, the oscillation energy $E_i(\omega)$ of node i can be written as

$$E_i(\omega) = \frac{1}{2}\,m_i\,\dot{x}_i(\omega,t)^2 + \frac{1}{2}\,m_i\,\omega^2\,x_i(\omega,t)^2. \tag{30}$$

If damping factor γ is sufficiently small, (30) is approximately equal to $E_i(\omega) \simeq \frac{1}{2}\,\omega^2 \sum_{\mu=0}^{n-1} A_\mu(\omega)^2\,v_\mu(i)^2$. In addition, we can consider that the contributions from eigenmodes other than eigenmode 0 are sufficiently small around $\omega \simeq \gamma/2$. Thus, we obtain

$$\begin{aligned} E_i\,(\gamma/2) &\simeq \frac{1}{2}\left(\frac{\gamma}{2}\right)^2 A_0\,(\gamma/2)^2\,v_0(i)^2 \\ &= \frac{\gamma^2}{8}\,A_0(\gamma/2)^2\left(\frac{\sqrt{m_i}}{\|\boldsymbol{v}_0\|}\right)^2 \quad \left(v_0(i)^2 \propto m_i\right) \\ &= C\,m_i, \end{aligned} \tag{31}$$

where constant C is independent of node i. Since the ratio of mass is more important than the magnitude of the value of mass, the mass can be obtained from (31) by removing the constant by a normalization condition or the like.

By substituting $c_{i,\mu}(\omega_\mu)$, m_i and m_j obtained by the above methods into (24) and (25), we can estimate the absolute value $|v_\mu(i)|$ ($\forall\mu$, i) of each component of all eigenvectors. However, in general, each eigenvector component has either positive or negative sign. Here we explain a method for determining the sign of eigenvector components by using the orthogonality $\boldsymbol{v}_\mu \cdot \boldsymbol{v}_\nu = \delta_{\mu\nu}$ of eigenvectors.

If we set $\sigma_{\mu i} \in \{-1, +1\}$ as the sign of $v_\mu(i)$, we define the objective function for each mode $\mu \neq 0$ as

$$\begin{aligned} Z_\mu &:= |\boldsymbol{v}_0 \cdot \boldsymbol{v}_\mu| = \left|\sum_{i=1}^{n} v_0(i)v_\mu(i)\right| \\ &= \left|\sum_{i=1}^{n} |v_0(i)v_\mu(i)|\sigma_{\mu i}\right| = \left|\sum_{i=1}^{n} r_{\mu i}\sigma_{\mu i}\right|, \end{aligned} \tag{32}$$

where $r_{\mu i} := |v_0(i)v_\mu(i)| \in \mathbb{R}$. Here we assume $\sigma_{0,i} = +1$ for all i, since all components of eigenvector $\boldsymbol{v}_0$ associated with the zero eigenvalue $\lambda_0 = 0$ have the same sign. (32) is a *number partitioning problem* (NPP) for real numbers; NPP is the task finding a partition of a set of positive integer such that the sums of elements in each subset are as close as possible. Determining the sequence $\boldsymbol{\sigma}_\mu = \{\sigma_{\mu,i}\}$ of signs that minimizes Z_μ of (32) is equivalent to finding the actual signs of components of eigenvector $\boldsymbol{v}_\mu$. Thus, it is expected that we can determine the signs of $\boldsymbol{v}_\mu$ by using the complete Karmarkar-Karp algorithm, which is a complete anytime algorithm for NPP. For more details, see Appendix.

Consequently, we can estimate the eigenvalues and eigenvectors of $\boldsymbol{S}$ even if the network structure is unknown. By substituting estimated eigenvalues and eigenvectors into (10), we can reproduce the scaled Laplacian matrix that represents the network topology and link weights. This means that we can estimate the network structure, which is virtually impossible to observe directly in the real world.

4. Numerical Results

In this section, through numerical experiments, we show that the network resonance method can estimate the structure of networks. Note that we assume that the amplitudes are given

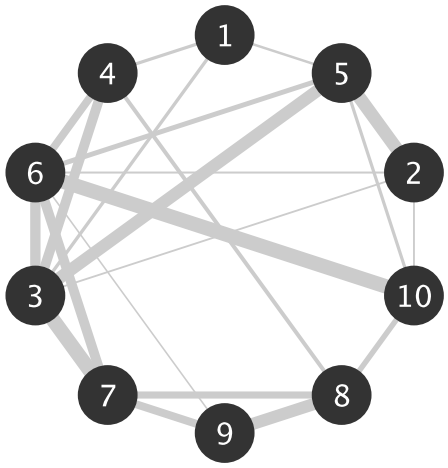

Fig. 5 Network used in the first experiment. Link width represents link weight.

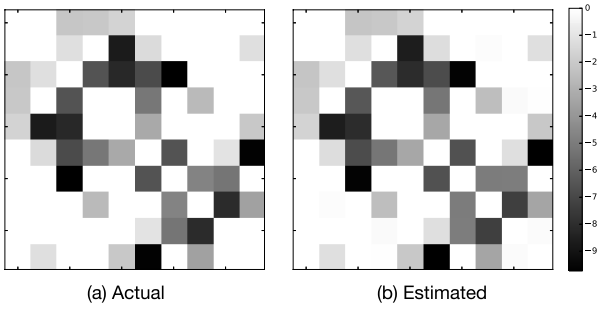

Fig. 6 Non-diagonal components of actual and estimated Laplacian matrix of Fig. 5 network.

in the simulation, since the displacement $x_i(t)$ is unobservable state value in our model, and we can only observe the amplitude $a_i(\omega)$ (or the oscillation energy $E_i(\omega) \propto a_i(\omega)^2$). Moreover, the frequency resolution is not constant but varied depending on the gradient $\mathrm{d}a_i(\omega)/\mathrm{d}\omega$; the external force frequency ω is a controlable parameter.

4.1 Estimation of the Undirected and Directed Network

First, for verifying the effectiveness of our method, we use the simple undirected network shown in Fig. 5. In this network, we set each link weight to a uniform random number in the interval $(1, 10)$. The mass of node i is 1 for all i, the strength F of the external force is 1, the damping factor $\gamma = 0.005$, the parameter k of (29) is 2. Moreover, we assume that the amplitude of all nodes are observable and we can input the external force into just nodes 1, 2 and 3.

The result of the first experiment is shown in Fig. 6. Figure 6(a) and 6(b) illustrate the non-diagonal components of the actual and estimated Laplacian matrix, respectively. It can be seen from this figure that our method can correctly estimate the presence and strength of links by observing the network's reaction. Indeed, the average value $\mathrm{ave}(e)$ and maximum value $\max(e)$ of relative errors $e_{ij} := |w_{ij} - \hat{w}_{ij}|/w_{ij}$ of non-zero components are small; $\mathrm{ave}(e) = 0.0293$ and $\max(e) = 0.1121$. Here the average value $\mathrm{ave}(e)$ is calculated by following equation:

$$\mathrm{ave}(e) = \frac{1}{|E|} \sum_{(i,j)\in E} e_{ij},$$

where E is the set of links of the network shown in Fig. 5.

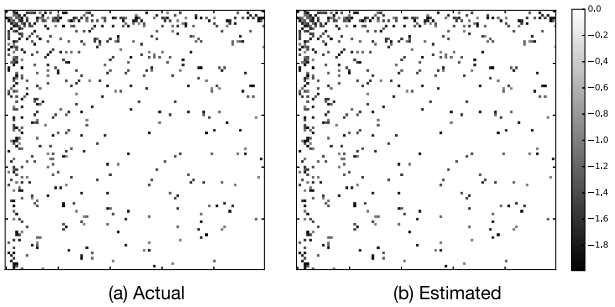

Fig. 7 Non-diagonal components of actual and estimated Laplacian matrix of a network generated by BA model.

Next, we demonstrate that our method can estimate large directed networks. In this experiment, we use a network generated by BA model [18]. BA model is a typical algorithm for generating random scale-free networks; it uses a preferential attachment mechanism, and requires three parameters (n, q_0, q): the number of nodes, the size of initial complete graph K_{q_0} and the number of links added at each step. Here the BA parameters are set to $(n, q_0, q) = (100, 3, 3)$. We set each link weight to a uniform random number in the interval $(1, 2)$, and node mass to a random integer from 1 to 10. Note that the asymmetric Laplacian matrix $\boldsymbol{\mathcal{L}}$ that satisfies Eqs. (5) and (6) can be encoded to the scaled Laplacian matrix $\boldsymbol{S}$ by the relation $\boldsymbol{S} = \boldsymbol{M}^{+1/2}\boldsymbol{\mathcal{L}}\boldsymbol{M}^{-1/2}$. Therefore, an undirected network whose nodes have unequal masses is interpreted as a directed network. Other parameters, F, γ and k, are the same as in the previous experiment. Moreover, we assume that the amplitude of all nodes are observable and we can input the external force into 50 nodes.

Figure 7 shows the non-diagonal components of the actual and estimated Laplacian matrices. From this figure, one recognizes that our method can also estimate a large directed network with high precision; $\mathrm{ave}(e) = 0.0005$ and $\max(e) = 0.0072$.

4.2 Relation between the Number of Accessible Nodes and Estimation Error

In this subsection, we investigate the relation between the number of accessible nodes and estimation errors under the same condition as Fig. 7 estimation; *accessible node* means the node where we can input the external force. Figure 8 shows the average and maximum value of relative errors versus the number of accessible nodes. This figure shows that the estimation errors decrease as the number of accessible nodes increases. Namely, our method can estimate a network structure with high precision if the number of accessible nodes is sufficiently large.

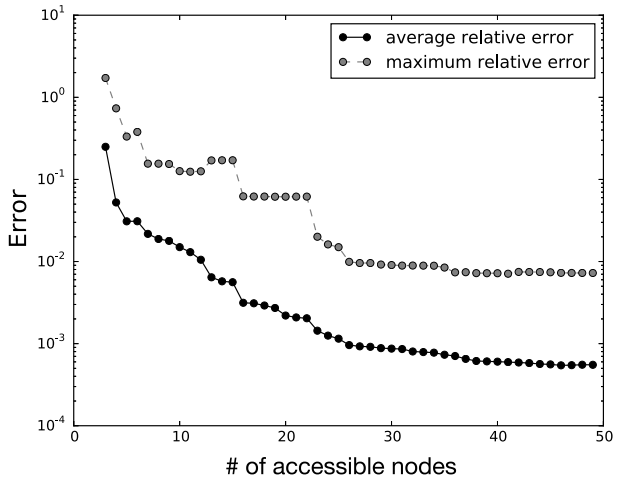

Fig. 8 Average and maximum value of relative errors versus the number of accessible nodes.

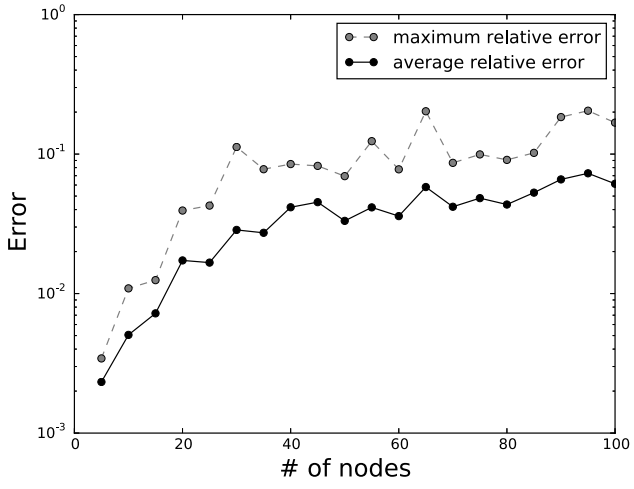

Fig. 9 Average and maximum value of relative errors versus the network size.

4.3 Relation between the Network Size and Estimation Error

In this subsection, we investigate the relation between the network size and estimation error. Here we generate an undirected BA networks with the parameters $(n, q_0, q) = (n, 3, 3)$ and set each link weight to a uniform random number in the interval (1,2). Other parameters are follows: $^{\forall i}\, m_i = 1$, $F = 1$, $\gamma = 0.001$ and $k = 2$. Then, the number of accessible nodes is $\lfloor 0.2n \rfloor$ for each network size n.

Figure 9 shows the average and maximum value of relative errors versus the network size. From this figure, we recognize that the estimation error tends to increase as the network size increases. Thus, our method cannot apply for too large networks and we need to restrict our method to a small network.

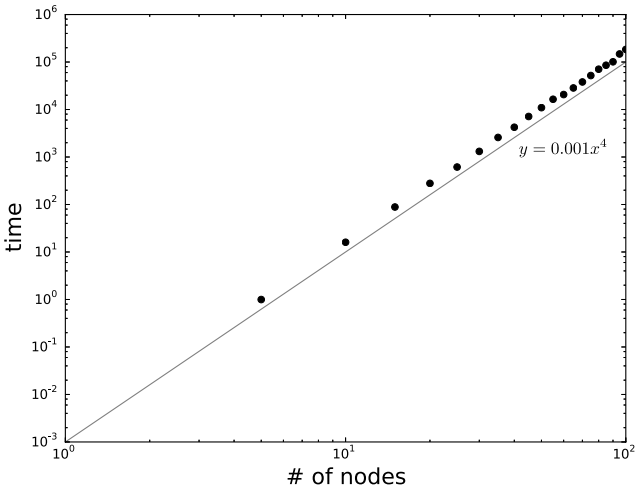

Fig. 10 The execution time of the processes *i* to *iii*.

4.4 Time Complexity

Finally, we show the time complexity of the proposed method. Our method can be classified roughly into four processes: *i) estimating eigenvalues, ii) estimating node masses, iii) estimating the absolute values of eigenvectors*, and *iv) determining the signs of eigenvectors*. Since the process *iv* is not deterministic, it is necessary to consider the time complexity separetely from the other processes. First, we show the execution time of the processes *i* to *iii* for the different network size (Fig. 10). Figure 10 shows that the time complexity of the processes *i* to *iii* is $O(n^4)$. Here the gray solid line is $y = 0.001x^4$ for comparison.

The time complexity of the process *iv* is discussed in [19]. According to [19], though the worst time complexity is $O(n2^n)$, the time complexity is reduced to $O(n^3)$ by the pruning rule of CKK, if network size is sufficiently large (around $n > 70$).

5. Discussion and Related Work

5.1 Discussion

In this subsection, we discuss the issues and the limitation of the network resonance method.

The first issue is duplication of eigenvalues. If eigenvalues of a scaled Laplacian matrix duplicate, we cannot estimate eigenvectors associated with duplicated eigenvalues. However, duplication of eigenvalues rarely occurs in the engineering framework. Eigenvalues of Laplacian marix $\boldsymbol{L}$ are generally calculated as the solutions of eigenvalue equation $\det(\lambda \boldsymbol{I} - \boldsymbol{L}) = (\lambda - \lambda_0)(\lambda - \lambda_1) \cdots (\lambda - \lambda_{n-1}) = 0$. Since the duplication of eigenvalues of Laplacian matrix $\boldsymbol{L}$ is caused by structural symmetry of a network [20], it is expected that the duplication of eigenvalues of Laplacian matrix rarely occurs in the case of the links of networks have weights. As an analogy, let us consider the condition that the quadratic equation has double roots. The quadratic equation $ax^2 + bx + c = 0$

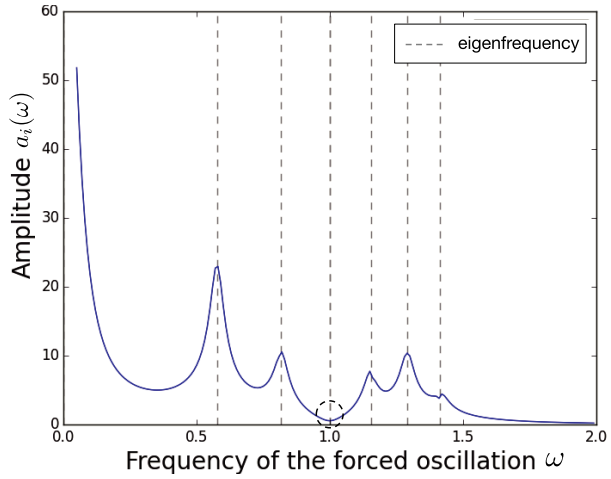

Fig. 11 An example of an amplitude with hidden peak. We cannot observe a peak around the eigenfrequency, since the absolute value of the component of the eigenvector is small (dashed circle).

has double roots iff $b^2 - 4ac = 0$. This situation rarely occurs if the coefficients $a, b, c \in \mathbb{R}$. Thus, we can theoretically observe all peaks by taking damping factor smaller than spacing between eigenvalues, i.e. $\gamma \ll \omega_{\mu+1} - \omega_\mu$. Although the damping factor depends on a network, we expect that it can be controled by coordinating the contents of input.

The second issue is hidden peaks. The height of peaks of amplitude $a_i(\omega)$ observed at node i depends on the magnitude of the product of $v_\mu(i)$ and $v_\mu(j)$, where j is the index of the node into which the external force is input. Thus, if the magnitude of $v_\mu(i)$ or $v_\mu(j)$ is rather small, the corresponding peak will not appear around the eigenfrequency (Fig. 11). In this case, we cannot estimate $|v_\mu(i)|$ because our method requires the information of peak height. An expedient solution is to change the node pair of the observing node i and the input node j. By changing the pair, the shape of the amplitude, $a_i(\omega)$, also changes. Thus, by repeating this trial-and-error approach to selecting good node pairs and finding peaks that are suitable, it is expected that $|v_\mu(i)|$ can generally be estimated. Moreover, as an effective method to overcome this issue, in [21], Sugimoto and Aida proposed a method that reproduces the Laplacian matrix from incomplete sets of eigenvalues and eigenvectors, i.e. only some of the eigenvalues and eigenvectors, by using compressed sensing. Thus, our method can estimate the structure of networks even if we cannot completely estimate all eigenvalues and eigenvectors.

The third isuue is the mutually interference of peaks. If the spacing between adjacent eigenfrequencies is extremely narrow, peaks mutually interfere and we may be prevented the eigenvector estimation. However, according to random matrix theory, *nerest neighbor spacing distribution* (NNSD), i.e. the distribution of the spacing between adjacent eigenvalues, of the Laplacian matrix of any networks follows *Gaussian orthogonal ensemble* (GOE) universality [22]. This means that the ratio of the region that the spacing is extremely narrow is constant, and it is expected that our method will not break down for large networks.

Then, we discuss the feasibility of our method. It is expected that our method is applicable to estimate the structure of various hidden networks; e.g. social networks of SNS users. In an example of application for social network of users, updating information on the web page with frequency ω at a certain node j corresponds to inputting the external force of the forced oscillation. Then, this influence is propagated through links of the network. Thus, by observing activity (oscillation energy) at each node i and estimating eigenvalues and eigenvectors of $\boldsymbol{S}$, we can estimate the structure of a social network. Note, in this example, damping factor γ of a network corresponds to user interest in contents of information. Thus, it is expected that we can control the damping factor γ such that it becomes sufficiently small by selecting its contents.

Finally, we discuss the limitation of our method. As suggested by Figs. 8 and 9, our method does not work well in the case that the network size is too large or we cannot access only one or few nodes. To overcome these limitations, we are currently investigating that the method for estimating network structure from only some of the eigenvalues and eigenvectors by using compressed sensing, based on that the sparsity of the structure of social networks.

5.2 Related Work

Estimating a network structure from dynamics on a network is a fundamental inverse problem. The *topology identification problem* is attracting attention, particularly in the field of nonlinear dynamics. Examples of existing solutions to this problem include an approach using synchronization [23], steady-state control [24], [25] and perturbation [26]. For details, see the survey [27]. These approaches assume nonlinear systems whose nodes are weakly coupled; e.g. Kuramoto model [28] or other phase oscillator systems. We, on the other hand, assume the oscillation model for describing network dynamics, and the oscillation model is a system whose nodes are strongly coupled. In this regard, our method differs from the above methods.

A few methods for estimating eigenvalues of a Laplacian matrix have been published. In [29], Franceschelli et al. proposed the method that estimates eigenvalues by artificially assigning local interaction dynamics to nodes and observing their response. Thus, this method can be applied only to network systems to which one can assign local interaction dynamics, such as ad-hoc network. In [30], [31], Mauroy and Hendrickx proposed the method that estimates eigenvalues by applying dynamic mode decomposition for snapshot data of observable state quantity in a network system. However, it seems that the method does not ensure estimating all eigenvalues of the Laplacian matrix of a large network. In contrast, our method can theoretically estimate all eigenvalues even if the size of a network is large, under the condition that the damping factor is sufficiently small.

Finally, we discuss the *frequency domain decomposition* (FDD) as the method for estimating eigenvalues and

eigenmodes of linear systems. FDD calculates *power spectral density* (PSD) from output (measurement) vectors to input signal such as the white noise, and estimates the eigenfrequencies and eigenmodes from peaks of PSD [32], [33]. Our method and FDD resemble in the respect that estimating eigenfrequencies and eigenmodes from peaks of response. However, as far as we investigated, there are some differences. First, FDD focuses on peaks of PSD, Fourier transform of auto-correlation function of a signal, while we focus on peaks of the amplitude versus external force frequencies. Although PSD and amplitude have common points that horizontal axis is frequency and peaks appears on specific frequencies, they are intrinsically different each other. Thus, FDD may not be directly applied to the dynamics that we assumed. Moreover, according to (19), the position of the maximal point of a peak shifts from the eigenfrequency a little. However, in the eigenfrequency estimation, it seems that FDD regards the position of the maximal point as the eigenfrequency, while our method considers this shift. Then, FDD sequentially estimates the eigenmodes (including signs), while our method firstly estimates the absolute value of eigenmodes (eigenvectors) and then determines the signs of eigenmodes. At this time, we estimate in the narrow area around peak to enhance the estimation accuracy. Although we cannot discuss superiority or inferiority because of difficulty of direct comparison, we believe that FDD does not spoil the originality of our method fatally.

6. Conclusion

In this paper, we proposed a network resonance method that can estimate network structure from the resonance of oscillation dynamics on networks. Our method can estimate the structure of undirected and some directed networks without *a priori* information about the presence and strength of links. Through numerical experiments, we showed the effectiveness of our method. Future work includes developing an efficient estimation method for larger and more complex networks and countermeasures to cases wherein the reactions of some nodes cannot be observed.

Acknowledgements

This research was supported by Grant-in-Aid for Scientific Research (B) No. 26280032 from JSPS.

References

[1] F.R. Chung, Spectral Graph Theory, American Mathematical Soc., 1997.
[2] D.J. Aldous, "Some inequalities for reversible markov chains," J. London Math. Soc., vol.2, no.3, pp.564–576, 1982.
[3] S. Boyd, P. Diaconis, P. Parrilo, and L. Xiao, "Fastest mixing markov chain on graphs with symmetries," SIAM J. Optimiz., vol.20, no.2, pp.792–819, 2009.
[4] M. Barahona and L.M. Pecora, "Synchronization in small-world systems," Phys. Rev. Lett., vol.89, no.5, p.054101, 2002.
[5] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D.U. Hwang, "Complex networks: Structure and dynamics," Physics Reports, vol.424, no.4, pp.175–308, 2006.
[6] A. Arenas, A. Diaz-Guilera, and C.J. Pérez-Vicente, "Synchronization reveals topological scales in complex networks," Phys. Rev. Lett., vol.96, no.11, p.114102, 2006.
[7] P.N. McGraw and M. Menzinger, "Analysis of nonlinear synchronization dynamics of oscillator networks by Laplacian spectral methods," Phys. Rev. E, vol.75, no.2, p.027104, 2007.
[8] V.M. Preciado and A. Jadbabaie, "Spectral analysis of virus spreading in random geometric networks," Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proc. 48th IEEE Conference on, pp.4802–4807, IEEE, 2009.
[9] P. Van Mieghem and J. Omic, "In-homogeneous virus spread in networks," arXiv preprint arXiv:1306.2588, 2013.
[10] M. Aida, C. Takano, and M. Murata, "Oscillation model for network dynamics caused by asymmetric node interaction based on the symmetric scaled Laplacian matrix," Proc. International Conference on Foundations of Computer Science (FCS), pp.38–44, The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp), 2016.
[11] M. Aida, C. Takano, and M. Murata, "Oscillation model for describing network dynamics caused by asymmetric node interaction," IEICE Trans. Commun., vol.E101-B, no.1, pp.123–136, Jan. 2018.
[12] S. Furutani, C. Takano, and M. Aida, "Proposal of the network resonance method for estimating eigenvalues of the scaled Laplacian matrix," Proc. International Conference on Intelligent Networking and Collaborative Systems (INCoS), pp.451–456, IEEE, 2016.
[13] S. Furutani, C. Takano, and M. Aida, "Method for estimating the eigenvectors of a scaled Laplacian matrix using the resonance of oscillation dynamics on networks," Proc. International Conference on Advances in social networks analysis and mining (ASONAM), pp.615–618, IEEE/ACM, 2017.
[14] M. Newman, Networks: An Introduction, Oxford University Press, 2010.
[15] Y. Li and Z.L. Zhang, "Digraph Laplacian and the degree of asymmetry," Internet Mathematics, vol.8, no.4, pp.381–401, 2012.
[16] C. Takano and M. Aida, "Proposal of new index for describing node centralities based on oscillation dynamics on network," Proc. Global Communications Conference (GLOBECOM), pp.1–7, IEEE, 2016.
[17] C. Takano and M. Aida, "Fundamental framework for describing various node centralities using an oscillation model on social media networks," Communications (ICC), 2017 IEEE International Conference on, pp.1–6, IEEE, 2017.
[18] A.L. Barabási and R. Albert, "Emergence of scaling in random networks," Science, vol.286, no.5439, pp.509–512, 1999.
[19] N. Hirakura, C. Takano, and M. Aida, "Efficient orthogonalizing the eigenvectors of the Laplacian matrix to estimate social network structure," NOLTA, pp.180–183, IEICE, 2018.
[20] B.D. MacArthur and R.J. Sánchez-García, "Spectral characteristics of network redundancy," Phys. Rev. E, vol.80, no.2, p.026117, 2009.
[21] S. Sugimoto and M. Aida, "Estimating the structure of social networks from incomplete set of observed information by using compressed sensing," Proc. IEEE Latin-American Conference on Communications (LATINCOM), IEEE, 2017.
[22] S. Jalan and J.N. Bandyopadhyay, "Random matrix analysis of network laplacians," Physica A: Statistical Mechanics and its Applications, vol.387, no.2-3, pp.667–674, 2008.
[23] D. Yu, M. Righero, and L. Kocarev, "Estimating topology of networks," Phys. Rev. Lett., vol.97, no.18, p.188701, 2006.
[24] D. Yu, "Estimating the topology of complex dynamical networks by steady state control: Generality and limitation," Automatica, vol.46, no.12, pp.2035–2040, 2010.
[25] D. Yu and U. Parlitz, "Driving a network to steady states reveals its cooperative architecture," Europhys. Lett., vol.81, no.4, p.48007, 2008.
[26] M. Timme, "Revealing network connectivity from response dynamics," Phys. Rev. Lett., vol.98, no.22, p.224101, 2007.
[27] M. Timme and J. Casadiego, "Revealing networks from dynamics:

An introduction," J. Phys. A: Math. Theor., vol.47, no.34, p.343001, 2014.
[28] Y. Kuramoto, Chemical Oscillations, Waves, and Turbulence, Springer Science & Business Media, 2012.
[29] M. Franceschelli, A. Gasparri, A. Giua, and C. Seatzu, "Decentralized estimation of Laplacian eigenvalues in multi-agent systems," Automatica, vol.49, no.4, pp.1031–1036, 2013.
[30] A. Mauroy and J. Hendrickx, "Spectral identification of networks using sparse measurements," SIAM J. Appl. Dyn. Syst., vol.16, no.1, pp.479–513, 2017.
[31] A. Mauroy and J. Hendrickx, "Spectral identification of networks with inputs," arXiv preprint arXiv:1709.04153, 2017.
[32] R. Brincker, L. Zhang, and P. Andersen, "Modal identification from ambient responses using frequency domain decomposition," Proc. 18th International Modal Analysis Conference (IMAC), 2000.
[33] R. Brincker, L. Zhang, and P. Andersen, "Modal identification of output-only systems using frequency domain decomposition," Smart Mater. Struct., vol.10, no.3, pp.441–445, 2001.
[34] S. Mertens, "The easiest hard problem: Number partitioning," Computational Complexity and Statistical Physics, vol.125, no.2, pp.125–139, 2006.
[35] I.P. Gent and T. Walsh, "Phase transitions and annealed theories: Number partitioning as a case study," ECAI, pp.170–174, PITMAN, 1996.
[36] J.P. Pedroso and M. Kubo, "Heuristics and exact methods for number partitioning," Eur. J. Oper. Res., vol.202, no.1, pp.73–81, 2010.
[37] R.E. Korf, "A complete anytime algorithm for number partitioning," Artif. Intell., vol.106, no.2, pp.181–203, 1998.

Appendix: Number Partitioning Problem

Number Partitioning Problem (NPP) is one of the classical NP-hard problems of combinatorial optimization, and it is actively studied in several fields such as mathematics, computer science and statistical physics [34]–[36]. NPP is defined easily: Given a set of positive numbers, $S = \{n_1, n_2, \ldots, n_N\}$, find a partition that minimizes the discrepancy

$$E = \left| \sum_{i \in U} n_i - \sum_{i \notin U} n_i \right|. \tag{A·1}$$

A partition that yields $E = 0$ or $E = 1$ is called a *perfect partition*.

A partition can be encoded by numbers $\sigma_i = \pm 1$: $\sigma_i = 1$ if $n_i \in U$ and $\sigma_i = -1$ otherwise. The cost function then reads

$$E = \left| \sum_{i=1}^{N} n_i \sigma_i \right|. \tag{A·2}$$

By comparing (32) and (A·2), we find (32) is an extension of (A·2). Thus, it is expected that we can determine the signs of components of eigenvectors by using the complete Karmarkar-Karp (CKK) algorithm [37], which is the *complete anytime algorithm* for NPP. The complete anytime algorithm is an algorithm that finds better and better solutions the longer it is allowed to run, until it finally finds and proves the optimum solution.

The procedure of the CKK algorithm is as follows: At each iteration, given the sequence $n_1, n_2, n_3, \ldots$ ($n_1 \geq n_2 \geq n_3 \geq \cdots$), CKK generates a new set by replacing the two largest numbers n_1 and n_2 by the absolute values of their difference. Moreover, CKK also generates a new set by replacing the two largest numbers n_1 and n_2 by their sum. This results in a binary tree, where the left branch replaces values by their difference, while the right branch replaces them by their sum:

$$\begin{cases} n_1, n_2, n_3, \cdots \mapsto n_1 - n_2, n_3, \ldots & \text{(left branch)}, \\ n_1, n_2, n_3, \cdots \mapsto n_1 + n_2, n_3, \ldots & \text{(right branch)}. \end{cases}$$

Iterating both operations $N - 1$ times generates a tree with 2^{N-1} terminal nodes. The terminal nodes are single element lists, whose elements show the discrepancy E of partitions. CKK searches this tree depth-first from left to right. Figure A·1 shows an example of a tree generated by CKK.

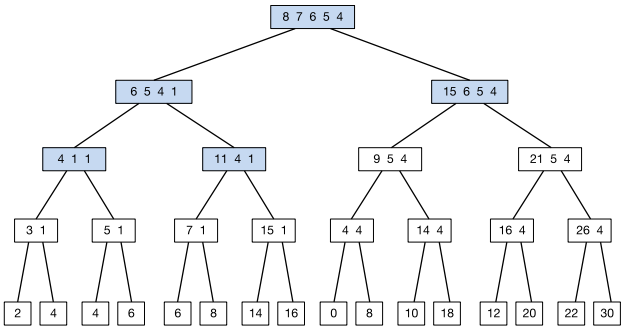

Fig. A·1 An example of a binary tree generated by the complete Karmarkar-Karp algorithm. Set $\{8, 7, 6, 5, 4\}$ can be split into two sets with the same summation of elements. With appropriate pruning rules, we have to visit only colored nodes to find the optimum solution.

The CKK algorithm may require exponential time in the worst case. However, it is possible to prune parts of the search tree by the following simple rules:

1. If less than 5 numbers are left in the set, take the left branch.
2. If the largest number in the set is larger than or equal to the sum of all the other numbers, stop branching: the best solution in this subtree is to place the largest number in one set and all the other numbers in the other set.
3. If a perfect partition $E = 0$ or $E = 1$ has been found, stop the whole algorithm.

From these rules, for large N, the number of searching node is nearly proportional to N [34]. Note that we need to alter the above stopping criteria $E = 0$ ($E = 1$) to $E < 10^{-k}$ ($k \in \mathbb{N}$) for determining the signs of eigenvectors.

By the way, we understand the concern that the misdetermination of signs of an eigenvector may prevent the estimation accuracy. However, we consider determination of the signs of eigenvectors have small effect for the estimation accuracy, since the objective function Z_μ of (32) takes 0 if and only if the sequence $\boldsymbol{\sigma}_\mu = \{\sigma_{\mu,i}\}$ of the actual signs

are found. Thus, in searching solution by CKK algorithm, we easily perceive that the sequence of signs is not actual if $Z_\mu \neq 0$. Therefore, by not using the eigenmodes that cannnot be determined in a suitable finite step, we can avoid a deteriotation of the estimation accuracy due to wrong signs of eigenvector.

Satoshi Furutani received his B.E. and M.E. degree in System Design Engineering from Tokyo Metropolitan University, Japan, in 2016 and 2018, respectively. Since joining Nippon Telegraph and Telephone Corporation (NTT) in 2018, he has been engaged in research and development of network security. His current interests include analysis of social network dynamics and mathematical modeling of cyber security dynamics.

Chisa Takano received a B.E. degree in Telecommunication Engineering from Osaka University, Japan, in 2000, and a Ph.D. in System Design Engineering from Tokyo Metropolitan University, Japan, in 2008. In 2000, she joined the Traffic Research Center, NTT Advanced Technology Corporation. Since April 2008, she has been an Associate Professor at the Graduate School of Information Sciences, Hiroshima City University. Her research interests are computer networks and distributed systems. She received the IEICE Young Researchers' Award in 2003. She received the Information Network Research Awards from the IEICE in 2004, 2012, and 2015. She is a member of IEEE (U.S.A.).

Masaki Aida received his B.S. degree in Physics and M.S. degree in Atomic Physics from St. Paul's University, Tokyo, Japan, in 1987 and 1989, respectively, and his Ph.D. in Telecommunications Engineering from the University of Tokyo, Japan, in 1999. In April 1989, he joined NTT Laboratories. From April 2005 to March 2007, he was an Associate Professor at the Faculty of System Design, Tokyo Metropolitan University. He has been a Professor of the Graduate School of Systems Design, Tokyo Metropolitan University since April 2007. His current interests include analysis of social network dynamics and distributed control of computer communication networks. He received the Best Tutorial Paper Award and the Best Paper Award of IEICE Communications Society in 2013 and 2016, respectively. He also received the IEICE 100-Year Memorial Paper Award in 2017. He is a member of the IEEE, ACM and the Operations Research Society of Japan.