$\mathcal{R}$-energy for Evaluating Robustness of Dynamic Networks

Ming Gao
Singapore Management University
minggao@smu.edu.sg

Ee-Peng Lim, David Lo
Living Analytics Research Center
Singapore Management University
{eplim,davidlo}@smu.edu.sg

ABSTRACT
The robustness of a network is determined by how well its vertices are connected to one another so as to keep the network strong and sustainable. As the network evolves, its robustness changes and may reveal events as well as periodic trend patterns that affect the interactions among users in the network. In this paper, we develop $\mathcal{R}$-energy as a new measure of network robustness based on the spectral analysis of normalized Laplacian matrix. $\mathcal{R}$-energy can cope with disconnected networks, and is efficient to compute with a time complexity of $O(|V| + |E|)$ where V and E are the vertex set and edge set of the network respectively. This makes $\mathcal{R}$-energy more efficient to compute than algebraic connectivity, another well known network robustness measure. Our experiments also show that removal of high degree vertices reduces network robustness (measured by $\mathcal{R}$-energy) more than that of random or small degree vertices. $\mathcal{R}$-energy can scale well for very large networks. It takes as little as 40 seconds to compute for a network with about 5M vertices and 69M edges. We can further detect events occurring in a dynamic Twitter network with about 130K users and discover interesting weekly tweeting trends by tracking changes to $\mathcal{R}$-energy.

Author Keywords
$\mathcal{R}$-energy; network robustness; normalized Laplacian matrix.

ACM Classification Keywords
G.2.2. Graph Theory: Spectra of graphs; Paths and connectivity problems

General Terms
Measurement; Performance

1. INTRODUCTION

The popularity of web, mobile phones and other portable devices has propelled the growth of large scale social networks such as Facebook and Twitter. These networks dynamically evolve as users join and leave, and as their interaction intensity changes over time. To characterize the strength of these large scale networks, we need some measures for their robustness.

WebSci'13, May 2–4, 2013, Paris, France.

The ability to measure the robustness of networks can benefit several useful applications. For example, in a phone call network, dense and frequent calls among users in the network reduce the likelihood of churn. The same can be observed for online social networks. Network robustness is also studied in other applications such as disease transmission [4][13], and network security [16], etc. For example, the robustness of IP networks affects service quality and security. Service providers therefore aim to monitor, manage and optimize their networks to keep their networks robust.

As today's networks are usually of very large scale, efficient measurement of network robustness is therefore a challenge. There are several previously proposed robustness measures in the research literature. They include: (a) *node connectivity* [12], (b) *edge connectivity* [12] and (c) *algebraic connectivity* [14]. Node (or edge) connectivity $\upsilon(G)$ (or $\varepsilon(G)$) of a network G is defined by the number of nodes (or edges) that may be removed to break the networks into multiple connected components. Large node and edge connectivity values suggest that a network is robust.

Algebraic connectivity $\lambda(G)$ is defined by the second smallest eigenvalue of the Laplacian matrix of the network. Algebraic connectivity is closely related to node connectivity and edge connectivity by the following inequality: $\upsilon(G) \leq \lambda(G) \leq \varepsilon(G)$ [14]. For example, both the node connectivity and edge connectivity of the network example in Figure 1(a) are 2. According to the above inequality, the algebraic connectivity of the network is therefore 2.

In combinatorics, an expander graph is a connected and undirected graph in which every small subset of the vertex set has a large boundary (to be explained shortly). The goodness (or robustness) of the expander graph can be measured by *vertex expansion* [5] and *edge expansion* [17]. Let $G = (V, E)$ be a connected and undirected network. The vertex expansion $h_v(G)$ and edge expansion $h_e(G)$ are defined in Equations 1 and 2 respectively.

$$h_v(G) = \min_{S \subset V, 0 < |S| \leq \frac{|V|}{2}} \frac{|\partial_{out}(S)|}{|S|} \quad (1)$$

$$h_e(G) = \min_{S \subset V, 0 < |S| \leq \frac{|V|}{2}} \frac{|\partial(S)|}{|S|} \quad (2)$$

where $\partial_{out}(S)$ is the outer vertex boundary of S (i.e., the set of vertices in $V \backslash S$ with at least one neighbor in S) and $\partial(S)$ is the edge boundary of S (i.e., the set of edges with exactly one endpoint in S). A robust expander graph contains fewer

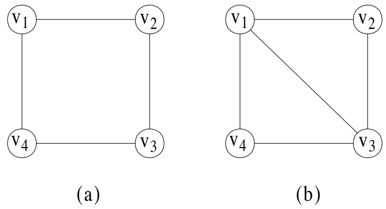


Figure 1: Two networks in the running example

graph	Connectivity			Expansion	
	node	edge	algebraic	vertex	edge
Figure 1(a)	2	2	2	1	1
Figure 1(b)	2	2	2	1	1.5

Table 1: Robustness metrics for networks in Figure 1

bridges or bottlenecks. Its vertex expansion and edge expansion measures are also related to algebraic connectivity by the Cheeger's Inequality [2]. Considering all cuts of the network in Figure 1(b), the vertex expansion and edge expansion are 1 and 1.5 respectively.

However, these measures have the following shortcomings:

- They are only applicable to connected networks. Even though a highly robust giant component exists in a network with very few connected components, the network is considered not robust at all as all these measures return zero values.

- They quantify robustness using specific (optimal) combinations of nodes (for node connectivity), specific combination of edges (for edge connectivity), and specific eigenvalue (for algebraic connectivity). For example, the network in Figure 1(b) is intuitively more robust than another network in Figure 1(a). However, the node, edge, algebraic connectivity and vertex expansion in Table 1 show that the robustness of the two networks are the same.

- They are difficult to scale for large networks of millions vertices and edges. Even though the edge expansion measure correctly tells us that the network in Figure 1(b) is more robust than that in Figure 1(a), it is computed by checking all cuts of the network, an expensive operation. For algebraic connectivity, we need to compute the second smallest eigenvalue of the Laplacian matrix. For node connectivity, edge connectivity and vertex expansion, we have to check all cuts of the network. These are all time consuming operations for large networks.

In this paper, we aims to address the problem of efficiently measuring network robustness which should be defined in a principled way. As the same network may contain one or more connected components over time, our network robustness should be able to cope with the dynamicity of network evolution. We summarize our contribution to the study of network robustness as follows:

- We propose $\mathcal{R}$-energy as an efficient measure for network non-robustness. Network robustness is thus the inverse of $\mathcal{R}$-energy. The new measure, defined based on normalized Laplacian matrix, demonstrates several nice properties. It can also handle networks with multiple connected components and can be computed with good time complexity $O(|V| + |E|)$, where V and E are node set and edge set of a network.

- We apply $\mathcal{R}$-energy to both synthetic and real networks. For a network with close to 5M vertices and 69M edges, the computation takes not more than 40 seconds. This shows that $\mathcal{R}$-energy can cope with large scale networks comfortably.

- We further apply $\mathcal{R}$-energy to a dynamic Twitter community with about 130K users to detect events and regular trend patterns that affect the network robustness. We empirically show that several significant events can be detected, and that users tend to be more active engaging each other on Sundays and Mondays, but not on Saturdays.

For the rest of the paper, we first give a brief survey of related work in Section 2. Next, we introduce some basic notations in Section 3 before presenting $\mathcal{R}$-energy and its algorithm in Section 4. In Section 5, we show some observations and the performance of $\mathcal{R}$-energy on both synthetic and real networks. In Section 6, we show some patterns and events found using $\mathcal{R}$-energy on a dynamic Twitter user community. Finally, we conclude this paper with directions for future research.

2. RELATED WORK

Robustness

The traditional network robustness measures, node connectivity and edge connectivity were proposed by Dekker and Colbert [12]. Graph expansion can also be used to measure network robustness. Different formulations of expander give rise to different measures of expander, e.g., edge expansion [17], vertex expansion [5] and spectral expansion [22]. Larger edge or vertex expansions indicate less bottleneck inside a network. Nevertheless, these measures do not work well for networks with multiple connected components.

Jamakovic and Mieghem proposed to use the second smallest eigenvalue of the Laplacian matrix also known as algebraic connectivity to measure network robustness [18, 14]. Malliaros, et al. described the relationship between algebraic connectivity and node/edge connectivities [22]. According to Cheeger's Inequality, Chung found that the expansion of a graph is closely related to the spectral gap between the largest and the second largest eigenvalues of adjacency matrix. Malliaros et al. confirmed the findings of Chung in [22]. This measure is however costly to compute and is sensitive to the network size. Hence, it is not appropriate for comparing networks of different sizes. Albert et al. used diameter to measure robustness of networks [1] but the measure does not capture network connectivity which should be considered in robustness measures.

Graph Energy

The energy of a graph G has always been defined to be some form of deviation of eigenvalues of some graph matrix from the mean of eigenvalues. For example, Gutman defined graph energy on an adjacency matrix as the absolute deviation of eigenvalues from the mean of eigenvalues which is zero for any adjacency matrix [15]. In [25, 27], *Laplacian energy* has been defined on the *combinatorial Laplacian matrix*. In [7], *normalized Laplacian energy* is defined on the *normalized Laplacian matrix* in a similar manner.

Day and So studied graph energy changes with edge or vertex removals [11][10]. There are some existing works which derive the lower and upper bounds for different energy definitions including Gutman's graph energy [3], Laplacian energy [25][27] [28] and normalized Laplacian energy [7]. They are not appropriate measures for network robustness of a graph as computing them would be time costly.

3. PRELIMINARIES

Our proposed $\mathcal{R}$-energy measure is based on *normalized Laplacian matrix*, random walk on graphs, and some other important concepts which will be covered in this section.

Normalized Laplacian

Consider an undirected graph $G = (V, E)$ with vertex set V and edge set E (Let $|V| = n$, and $|E| = m$). Let A_G denote the adjacency matrix representing G and is defined as:

$$A_G(i,j) := \begin{cases} 1, & \text{if } (v_i, v_j) \in E; \\ 0, & \text{otherwise.} \end{cases}$$

The degree of a vertex v, $d(v)$, and degree matrix D_G can be defined as:

$$d(v) := |\{u|(v,u) \in E\}| \quad (3)$$

$$D_G(i,j) := \begin{cases} d(v_i), & \text{if } i = j \text{ and } v_i \in V; \\ 0, & \text{otherwise.} \end{cases} \quad (4)$$

Based on matrixes A_G and D_G, we define the **normalized Laplacian matrix** in Definition 1.

DEFINITION 1. *The **normalized Laplacian matrix** N_G of a graph G with nonnegative adjacency matrix A_G is given by*

$$N_G := I - D_G^{-1/2} A_G D_G^{-1/2}.$$

We denote

$$0 = \zeta_1 \le \zeta_2 \le \cdots \le \zeta_n \quad (5)$$

as the eigenvalues of the normalized Laplacian matrix N_G.

There are several important properties about the eigenvalues of normalized Laplacian matrix as presented in Lemma 1.

LEMMA 1. *The eigenvalues of the normalized Laplacian matrix of a graph with n vertices satisfy the following properties:*

(i) $0 \le \zeta_2 \le \frac{n}{n-1} \le \zeta_n \le 2$*, for* $i = 1, 2, \cdots, n$*;*

(ii) $\zeta_2 = \cdots = \zeta_n = \frac{n}{n-1}$ *if and only if G is a clique;*

(iii) $\zeta_n = 2$ *if and only if G is a biclique;*

(iv) G has at least i connected components if and only if $\zeta_j = 0$*, for* $j = 1, 2, \cdots, i$*.*

Property (i) says that the eigenvalues of normalized Laplacian matrix range from 0 to 2. As a special case, when all except the smallest eigenvalue equal $\frac{n}{n-1}$, the graph is a clique as shown in Property (ii). Property (iii) states that the largest eigenvalue takes the upper bound value 2, the graph will be a biclique. A network therefore resemble a biclique when the largest eigenvalue is close to 2, or a clique when many eigenvalues are close to $\frac{n}{n-1}$. Property (iv) states that each additional connected component corresponds to having the next smaller eigenvalue assigned with a zero value. ζ_1 is therefore 0 in any network.

Normalized Laplacian matrix is important when modeling random walk on a graph. We thus introduce the relationship between normalized Laplacian matrix and random walk next.

Random Walk

Given a graph and a starting vertex, we select a neighbor of it at random, and move to this neighbor. From there, we repeat the step by selecting another neighbor at random, and move to it and so on. The random sequence of vertices visited in this way is known as a *random walk* on the graph.

Let $P = (p_{ij})_{1 \le i,j \le n}$ denote the transition probabilities of random walk, where p_{ij} denotes the probability of visiting v_j from v_i in one step. So

$$p_{ij} := \begin{cases} \frac{1}{d(v_i)}, & \text{if } (v_i, v_j) \in E; \\ 0, & \text{otherwise.} \end{cases}$$

In the matrix form, transition probability matrix can be defined as:

$$P = D_G^{-1} \cdot A_G.$$

Furthermore, the rule of random walk can be expressed by the simple equation:

$$W^t = P \cdot W^{t-1} = p^2 \cdot W^{t-2} = \cdots = P^t \cdot W^0, \quad (6)$$

where W^0 represents the initial state of the random walk and W^t represents the state after t steps.

Nevertheless, P is not a symmetric matrix and this complicates its eigenvalue analysis. An important fact is that the eigenvalues of normalized Laplacian matrix N_G are closely related to the eigenvalues of P. With this, we can use the $N(G)$ to analyze spectrum for random walker. Formally,

LEMMA 2. *Let $\zeta_1, \cdots, \zeta_n$'s be the eigenvalues of $N(G)$. P have the eigenvalues equal to $1 - \zeta_i$, for $i = 1, 2, \cdots, n$.*

PROOF. Note that

$$N_G = I - D_G^{-1/2} A_G D_G^{-1/2} = D_G^{1/2}(I - P)D_G^{-1/2}.$$

Suppose that v is an eigenvector of N_G, with eigenvalue ζ. We have

$$N_G \cdot v = \zeta \cdot v$$

Let $q = D_G^{-1/2} \cdot v$, then,

$$\zeta \cdot v = D_G^{1/2}(I - P)D_G^{-1/2} \cdot v = D_G^{1/2}(I - P) \cdot q \quad (7)$$

By multiplying by $D_G^{-1/2}$ to Equation 7, we obtain

$$(I - P) \cdot q = \zeta \cdot D_G^{-1/2} \cdot v = \zeta \cdot q.$$

Therefore, q is an eigenvector of $I - P$ with eigenvalue ζ. □

2-step commute probability

Based on the spectral analysis of normalized Laplacian matrix, our proposed robustness metric is closely related to *2-step commute probability* of vertices in the random walk (elaborated in next section).

In Equation 6, each entry p_{ij}^t of P^t is the probability that, starting at v_i, the walker reaches v_j in t steps. Specially, the entry p_{ii}^2 represents the probability of reaching v_i from v_i in exactly 2 steps. This is also known as the **2-step commute probability** of vertex v_i. Computationally,

$$p_{ii}^2 = \sum_{j=1}^{n} p_{ij} \cdot p_{ji}$$

2-step commute probability is very important as it measures the possibility of a random walk returning to vertex v_i after 2 steps. A lower value suggests that the walker starting from v_i is less likely to return in 2 steps, but more likely to visit the other vertices.

EXAMPLE 1. *Figures 1(a) and 1(b) show two undirected graphs with 4 vertices in a circle. Thus, the transition probability matrixes can be derived as*

$$P = \begin{pmatrix} 0 & \frac{1}{2} & 0 & \frac{1}{2} \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 \\ 0 & \frac{1}{2} & 0 & \frac{1}{2} \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 \end{pmatrix}, Q = \begin{pmatrix} 0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 \\ \frac{1}{3} & \frac{1}{3} & 0 & \frac{1}{3} \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 \end{pmatrix}$$

We then compute the 2-step commute probabilities of vertex v_3 as

$$p_{33}^2 = \sum_{j=1}^{4} p_{3j} \cdot p_{j3} = 0 \cdot 0 + \frac{1}{2} \cdot \frac{1}{2} + 0 \cdot 0 + \frac{1}{2} \cdot \frac{1}{2} = \frac{1}{2}$$

$$q_{33}^2 = \sum_{j=1}^{4} p_{3j} \cdot p_{j3} = \frac{1}{3} \cdot \frac{1}{3} + \frac{1}{3} \cdot \frac{1}{2} + 0 \cdot 0 + \frac{1}{3} \cdot \frac{1}{2} = \frac{4}{9}$$

Comparing the two 2-step commute probabilities, we know that v_3 in Figure 1(b) can reach the other vertices more easily than that in Figure 1(a) which follows our intuition.

4. $\mathcal{R}$-ENERGY

In this section, we give the definition of $\mathcal{R}$-energy and present an efficient approach to compute $\mathcal{R}$-energy.

$\mathcal{R}$-energy

According to Lemma 1, for a network G that is sparsely connected and is far from being a clique, its ζ_2 is small but ζ_n is large. In contrast, a network that are densely connected and similar to a clique will have ζ_2 not much smaller than ζ_n. In other words, a robust network should have a small gap between ζ_2 and ζ_n. This gap between ζ_2 and ζ_n can therefore be used to measure network robustness.

In statistics, the range of a set of data values is measured by variability, which is defined by the gap between the largest and the smallest data values [26]. For networks, we may use some variability measures over their eigenvalue sequence $\zeta_2, \zeta_3, \cdots, \zeta_n$. Recall that ζ_1 is always zero. Examples of such variability measures include variance, standard deviation and relative variability [26]. In the following definition, we use the variance of the eigenvalues to measure network robustness and call the new measure **robustness energy**.

DEFINITION 2. *Let G be a network. The **robustness energy** (or **$\mathcal{R}$-energy**) of G is defined as:*

$$\mathbb{E}_{\mathcal{R}}(G) := \frac{1}{n-1} \sum_{i=2}^{n} (\zeta_i - \overline{\zeta})^2$$

where $\overline{\zeta} = \frac{1}{n-1} \sum_{i=2}^{n} \zeta_i$.

$\mathcal{R}$-energy is always non-negative and the smaller is $\mathcal{R}$-energy, the more robust is the network. This is because smaller variability of $\zeta_2, \zeta_3, \cdots, \zeta_n$ implies that the network is closer to a clique.

$\mathcal{R}$-energy can be used to measure the robustness of both connected and disconnected networks. Networks with multiple connected components will see larger variability of their eigenvalues (as more ζ_i's are zeros) leading to larger $\mathcal{R}$-energy values.

The naive approach to compute $\mathcal{R}$-energy after obtaining all eigenvalues of the normalized Laplacian matrix is computationally expensive. In the next subsection, we therefore analyze the spectrum of normalized Laplacian matrix, and propose a simple and efficient approach to compute $\mathcal{R}$-energy in $O(|V| + |E|)$ time complexity.

Computation of the $\mathcal{R}$-energy

To compute the variance of eigenvalues $\zeta_2, \zeta_3, \cdots, \zeta_n$, we compute the mean using Theorem 1.

THEOREM 1. *The mean of eigenvalues $\zeta_2, \zeta_3, \cdots, \zeta_n$ of a network with n vertices, $\overline{\zeta}$, is $\frac{n}{n-1}$.*

PROOF. According to linear algebra, the trace of a matrix is defined by the sum of all its diagonal elements, and this is also equal to the sum of all eigenvalues of the matrix. Recall Definition 1, each entry $N_G(i,j)$ of N_G (i.e., normalized Laplacian matrix) is as follows:

$$N_G(i,j) = \begin{cases} 1, & \text{if } i = j \text{ and } d(v_i) \neq 0; \\ -\frac{1}{\sqrt{d(v_i)d(v_j)}}, & \text{if } A_G(i,j) \neq 0; \\ 0, & otherwise. \end{cases} \quad (8)$$

Each diagonal element of N_G is therefore 1. Note that $\zeta_1 = 0$. Thus,

$$\frac{1}{n-1}\sum_{i=2}^{n}\zeta_i = \frac{1}{n-1}\sum_{i=1}^{n}\zeta_i = \frac{1}{n-1}\cdot tr(N_G) = \frac{n}{n-1}.$$

where $tr(N_G)$ denotes the trace of matrix N_G. □

With the mean of the eigenvalue sequence, we can now compute $\mathcal{R}$-energy using Theorem 2.

THEOREM 2. *The $\mathcal{R}$-energy of G satisfies Equation 9.*

$$\mathbb{E}_{\mathcal{R}}(G) = \frac{1}{n-1}\sum_{i=1}^{n}\sum_{j\neq i}^{n}\frac{A_G(i,j)}{d(v_i)d(v_j)} - \frac{n}{(n-1)^2}. \quad (9)$$

PROOF. According to Theorem 1, the $\mathcal{R}$-energy of G can be expressed as:

$$\begin{aligned}\mathbb{E}_{\mathcal{R}}(G) &= \frac{1}{n-1}\sum_{i=2}^{n}(\zeta_i - \frac{n}{n-1})^2 \\ &= \frac{1}{n-1}\sum_{i=2}^{n}\zeta_i^2 - \frac{2n}{(n-1)^2}\sum_{i=2}^{n}\zeta_i + \frac{n^2}{(n-1)^2} \\ &= \frac{1}{n-1}\sum_{i=1}^{n}\zeta_i^2 - \frac{n^2}{(n-1)^2}\end{aligned}$$

According to Equation 8, the i^{th} diagonal element of $N(G)^2$ is

$$\sum_{j=1}^{n} N_G(i,j)N_G(j,i) = \sum_{j\neq i}^{n}\frac{A_G(i,j)}{d(v_i)d(v_j)} + 1.$$

Applying an important property of the trace, i.e., $\sum_{i=1}^{n}\zeta_i^2 = tr(N(G)^2)$, we obtain the following:

$$\mathbb{E}_{\mathcal{R}}(G) = \frac{1}{n-1}\sum_{i=1}^{n}\sum_{j\neq i}^{n}\frac{A_G(i,j)}{d(v_i)d(v_j)} - \frac{n}{(n-1)^2}$$

□

Interestingly, the first term of the above $\mathbb{E}_{\mathcal{R}}(G)$ equation represents the average 2-step commute probability of vertices in G. Recall that the 2-step commute probability of all vertices can be computed as:

$$\frac{1}{n}\sum_{i=1}^{n}p_{ii}^2 = \frac{1}{n}\sum_{i=1}^{n}\sum_{j\neq i}^{n}p_{ij}\cdot p_{ji} = \frac{1}{n}\sum_{i=1}^{n}\sum_{j\neq i}^{n}\frac{A_G(i,j)}{d(v_i)d(v_j)},$$

Equation 9 can be further simplified to Equations 10 and 11.

$$\begin{aligned}\mathbb{E}_{\mathcal{R}}(G) &= \tfrac{n}{n-1}\big(\tfrac{1}{n}\textstyle\sum_{i=1}^{n}\sum_{j\neq i}^{n}\tfrac{A_G(i,j)}{d(v_i)d(v_j)} - \tfrac{1}{n-1}\big) \quad (10)\\ &= \tfrac{1}{n-1}\textstyle\sum_{(v_i,v_j)\in E}\tfrac{1}{d(v_i)d(v_j)} - \tfrac{n}{(n-1)^2} \quad (11)\end{aligned}$$

The factor $\frac{n}{n-1}$ in Equation 10 can be considered as a reward factor for the network of n vertices. Larger graphs are therefore more robust due to monotonically decreasing $\frac{n}{n-1}$ as n increases. This factor facilitates the comparison of $\mathcal{R}$-energy for networks with different sizes.

Note that the 2-step commute probability of a clique with n vertices is $\frac{1}{n-1}$. The right side of Equation 10 is thus the difference between the average 2-step commute probability and the 2-step community probability of a clique with the same size. Hence, the $\mathcal{R}$-energy of G combines the reward of network size with the difference between the 2-step commute probability of G and that of a clique.

As $\mathcal{R}$-energy can be expressed by Equation 11, we can efficiently compute it by scanning all edges of the network after computing the degrees of vertices in the network. Algorithm 1 depicts the steps to compute the $\mathcal{R}$-energy of a network. The algorithm consists of two main steps. One is to compute the degree of vertices (Lines 1-3). Another is to aggregate the 2-step commute probabilities of vertices at Lines 4-6. The $\mathcal{R}$-energy is finally obtained at Line 7. Both the time and space complexities of the algorithm are $O(|V| + |E|)$.

Using Algorithm 1, the $\mathcal{R}-$energies of networks shown in Figures 1(a) and 1(b) are 0.222 and 0.074 respectively. The $\mathcal{R}$-energy values of the networks match our intuition that the former is more robust than the latter. Compared to other measures shown in Table 1, our proposed $\mathcal{R}$-energy measure is better than most existing robustness measures (except for edge expansion).

Similar to the existing normalized Laplacian energy [7], we can also define a robustness measure using absolute deviation of eigenvalue sequence $\zeta_2, \zeta_3, \cdots, \zeta_n$. The measure however cannot be computed efficiently.

$\mathcal{R}$-energy can measure the robustness of both connected and disconnected graphs. Suppose that network G has N connected components, denoted as $\{C_k\}_{k=1}^{N}$. In Equation 12, the energy is derived by weighted sum of the average 2-step commute probability of vertices from each connected component.

$$\mathbb{E}_{\mathcal{R}}(G) = \frac{n}{n-1}(\sum_{k=1}^{N}\frac{n_k}{n}P_{C_k} - \frac{1}{n-1}) \quad (12)$$

where P_{C_k} is the average 2-step commute probability of vertices from connected component C_k in Equation 13.

$$P_{C_k} = \frac{1}{n_k}\sum_{(v_i,v_j)\in C_k}\frac{A_G(i,j)}{d(v_i)d(v_j)}, k = 1, \cdots, N \quad (13)$$

$\mathcal{R}$-energy therefore considers a large disconnected network G to be robust if G contains a robust giant component.

5. ROBUSTNESS OF LARGE STATIC NETWORKS

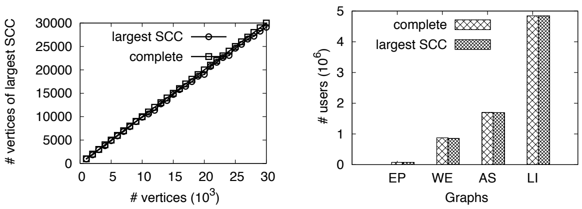
In this section, we evaluate our proposed $\mathcal{R}$-energy and other robustness measures on static networks including synthetically created networks and some real world networks. We design a set of experiments to compare the effectiveness and scalability of $\mathcal{R}$-energy with algebraic connectivity. We also seek to find any common patterns that appear in the networks. The experiments on synthetic networks were implemented in Matlab while those on real networks were implemented in Java. They were all conducted on a dual core 64-bit processor with 3.06 GHz CPUs and 128GB of RAM.

ALGORITHM 1: $calEnergy(G)$

Input: input graph: $G = (V, E)$;
Output: the $\mathcal{R}$-energy of G: e;
for *each vertex* $v \in V$ **do**
  $deg(v) = |\{u|(v, u) \in E\}|$;
end
for *each edge* $(v, u) \in E$ **do**
  $e \leftarrow e + \frac{1}{deg(v)deg(u)}$;
end
$e \leftarrow \frac{e}{n-1} - \frac{n}{(n-1)^2}$;
return e

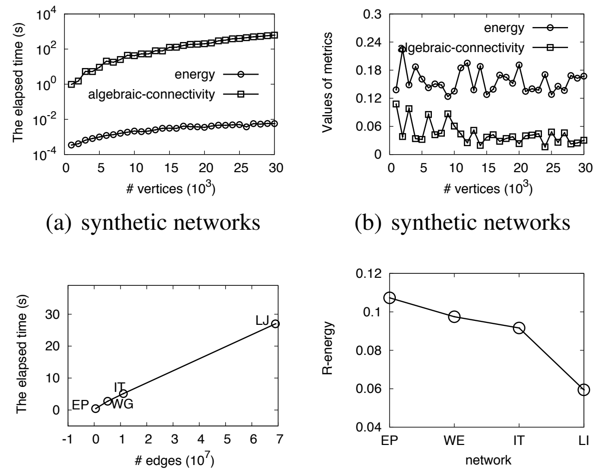
ID	network name	vertices	edges	density
EP	Epinion [8]	75,879	508,837	1.767×10^{-4}
WG	Web graph [21]	875,713	5,105,039	1.331×10^{-5}
IT	Internet topology [20]	1,696,415	11,095,298	7.710×10^{-6}
LJ	LiveJournal [21]	4,846,609	68,475,391	5.830×10^{-6}

Table 2: Descriptive statistics of real networks



(a) synthetic networks (b) real networks

Figure 2: Size of largest SCCs (giant components)



(c) real networks (d) real networks

Figure 3: Performance of computing the $\mathcal{R}$-energy

Networks

Synthetic networks. We generate different synthetic networks of N ($1K \leq N \leq 30K$) vertices with power law degree distribution using the graph generation algorithm proposed by Palmer and Steffan [24]. Each synthetic network with N vertices is denoted by Syn_N. In essence, the algorithm starts with a graph with N vertices but no edges. It then assigns a degree k to each vertex v such that $Pr[deg(v) = k] \approx k^{-\alpha}$ where α is the exponent > 2. Finally, the vertices are processed in decreasing degree order, and are assigned neighbors according to their degrees.

Real networks. We use four static real networks with different sizes from Stanford Large Network Dataset Collection, namely Epinions, Web graph, Internet topology and LiveJournal [1]. The descriptive statistics of these networks are shown in Table 2. In this work, we consider these networks unweighted and undirected.

To evaluate the algebraic connectivity, we extract the largest connected component and use it as a representative of the entire graph as there is always a giant component in each of these networks [9]. Figures 2(a) and 2(b) confirm the giant components in both synthetic and real networks.

Efficiency and scalability of $\mathcal{R}$-energy

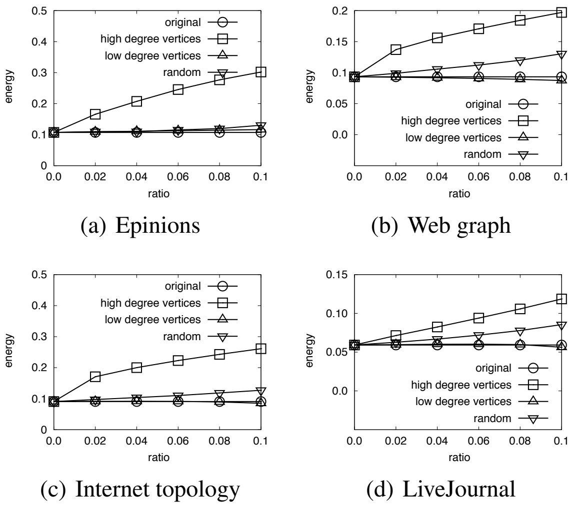
Figure 3(a) illustrates the *elapsed time* of computing both $\mathcal{R}$-energy and algebraic connectivity on synthetic networks. We observe that Matlab takes almost one hour to compute the algebraic connectivity of the network with $30,000$ vertices[2]. On the other hand, the elapsed time of $\mathcal{R}$-energy computation for the same network is more than 4 order of magnitude faster than that of the algebraic connectivity for a large network. We further observe that the elapsed time for $\mathcal{R}$-energy scales linearly with the number of edges making it ideal for networks with millions vertices.

We evaluate the correlation between $\mathcal{R}$-energy and algebraic connectivity on the different synthetic networks to see if they produce similar robustness results. Note that small $\mathcal{R}$-energy suggests high robustness, while high algebraic connectivity suggests high robustness. As shown in Figure 3(b), the two measures are negatively correlated with Pearson correlation coefficient -0.62. This shows that the correlation between them is strong (< -0.5).

Figure 3(c) depicts the elapsed time of computing $\mathcal{R}$-energy and algebraic connectivity for four real networks. The elapsed time for algebraic connectivity for the smallest real network, i.e., Epinion, is more than 12 hours. For the largest network LiveJournal [21] with 4.8M vertices and 68.5M edges, $\mathcal{R}$-energy takes less than 40 seconds to compute. Figure 3(b) shows $\mathcal{R}$-energies for these real networks.

Impact of vertex removal to $\mathcal{R}$-energy

[1]http://snap.stanford.edu/data/index.html



(a) Epinions (b) Web graph

(c) Internet topology (d) LiveJournal

Figure 4: $\mathcal{R}-$energies of static graphs

Complex networks with heavy-tail are known to be highly robust against random removal of vertices [9], but are hypersensitive to removal of high degree vertices [1][6]. We would like to verify this using $\mathcal{R}$-energy measure.

We experiment with three vertex removal options, namely, (a) remove in decreasing degree order; (b) remove in increasing degree order; and (c) remove in random order. For each option, after removing x fraction of vertices from the largest connected component, we compute $\mathcal{R}$-energy to measure the new network robustness. Figure 4 illustrates the $\mathcal{R}$-energy of resultant network for the three options compared with the $\mathcal{R}$-energy of the original network. From the figure, we obtain three important observations as follows.

OBSERVATION 1. *Networks become less robust sooner when vertices of the highest degrees are removed.*

Compared with the original graph, $\mathcal{R}$-energy increases sooner when vertices of the highest degrees are removed than when vertices of random degrees or small degrees are removed. This is the case since vertices of high degrees tend to have smaller 2-step commute probabilities. Removing them leads to an increase in average probability. Therefore, the network becomes less robust.

OBSERVATION 2. *Networks remain robust or become slightly more robust when vertices of the smallest degrees are removed.*

Figure 4 shows that $\mathcal{R}$-energy remains constant or decreases slightly when vertices of the smallest degrees are removed from the network. Again, the smallest degree vertices have larger 2-step commute probabilities. Removing the smallest degree vertices results in little decrease in the average of the probabilities.

[2] We employ Matlab function $eigs(L, 2, -1.0)$ to compute two eigenvalues which are closest to -1, i.e., the smallest and the second smallest eigenvalues of L, where L is a sparse matrix.

OBSERVATION 3. *Networks become less robust when vertices are randomly removed. However, the change is slower than that of removing vertices of highest degrees.*

This observation can be attributed to the fact that each vertex has a certain chance to decrease its degree when we remove vertices at random. That means the 2-step commute probability of each vertex increases with certain probability. However, vertices of smallest degree are more likely to be removed in scale-free networks. Hence, vertices of large 2-step commute probabilities are more likely to be removed leading to a decrease in graph energy.

The above three observations are also consistent to the results of Albert et al.'s work which uses diameter to measure robustness of networks [1]. They found that scale-free networks are robust to random vertex removals, but not to removals of most connected vertices. The diameter of a network decreases when vertices of the smallest degrees are removed and the network becomes highly robust.

6. DETECTING EVENTS AND TRENDS USING ROBUSTNESS

Networks evolve with time and so are their robustness. In this section, we apply $\mathcal{R}$-energy on dynamic and time-evolving Twitter network so as to evaluate robustness as a possible measure to detect events and trends. Before that, we use the number of replies or retweets to detect events. A few events are found due to intense fluctuation in these time series. Unlike the previous event and trend detection research which considers time series of messages or news articles generated in social media, our approach utilizes dynamic changes to network structure. These are the changes that cause a network to become suddenly more robust or less robust than usual.

Data collection

Twitter is a popular microblogging site with users generating and sharing short message contents in real time [19]. In this experiment, we first selected a set of Twitter users U who are the followers and followees of a small set of seed user accounts that belong to US politicians and analysts. These are the users who are more likely to tweet about political topics. We crawled the Twitter data generated by U from 1^{st} May, 2012 to 29^{th} July, 2012.

From U, we further selected users who write, reply or retweet at least a tweet per month over three months. There are 129,056 such users and we keep them in the user set U discarding the remaining users and their tweets. Each day, a subset of users in U may reply or retweet one another. We therefore construct a *reply network* and another *retweet network* for day t and denote them by $G^{RE}(t)$ and $G^{RT}(t)$ respectively. An undirected edge (u, v) is included in the reply network for day t if user u replies at least a tweet from user v, or user v replies at least a tweet from user u in day t. The edges in retweet network on day t are created in a similar manner.

Event detection

We show the $\mathcal{R}-$energies of $G^{RE}(t)$ and $G^{RT}(t)$ in Figures 5(a) and 5(b). To facilitate reading, we add vertical lines

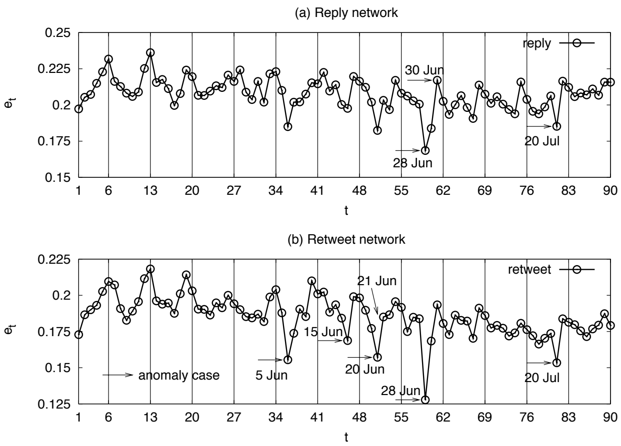


Figure 5: $\mathcal{R}$-energy on G^{RE} and G^{RT} (Note: Vertical lines denote Sundays)

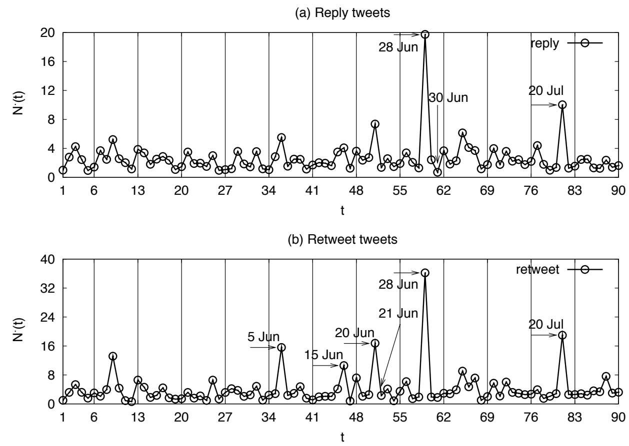


Figure 6: Normalized difference of word frequencies on G^{RE} and G^{RT} (Note: Vertical lines denote Sundays)

representing Sundays to the figures. From the figures, we aim to determine events that are characterized by bursts and drops of communication (replies or retweets) by many users. We call these the *internal* and *external events* as the former can be explained by the bursty content but not the latter. For example, a sport event may draw user attention away from tweeting about politics. In addition to event detection, we also want to explain internal events by searching the web.

Suppose $(e_1, e_2 \cdots, e_{90})$ is the sequence of $\mathcal{R}$-energy values. We calculate the absolute first-order difference of energy sequence, denoted as $(d_1, d_2 \cdots, d_{90})$, where $d_1 = 0$ and $d_{t+1} = |e_{t+1} - e_t|$ for $1 \leq t \leq 89$. Based on the mean and standard deviation of $\{d_t\}$, we can detect an event at time t' statistically if $|d_{t'} - mean(\{d_t\})| > \gamma \cdot stddev(\{d_t\})$ where $mean(\{d_t\})$ and $stddev(\{d_t\})$ denote the mean and standard deviation of $\{d_t\}$ respectively. In other words, an event is found when the absolute first-order difference deviates from mean more than γ times the standard deviation. However, mean is known to be sensitive to anomalies. We therefore employ trimmed mean that is defined as the mean after discarding the smallest and largest $\tau\% \cdot d_t$ values. In this work, we set $\gamma = 3$ and $\tau = 5$ empirically. Selecting values for these parameters is an interesting research but we shall leave this to our future work.

To describe an event at day t, we need to extract relevant event description keywords from tweets (which can be replies or retweets) generated on the same day t. We denote the words extracted from reply tweets (or retweets) on day t by $W^{RE}(t)$ (or $W^{RT}(t)$) and the frequency of word $w \in W^{RE}(t)$ (or $W^{RT}(t)$) by $f^{RE}(w, t)$ (or $f^{RT}(w, t)$). We define the first-order frequency difference of word w for day t as $df^*(w, t) = f^*(w, t) - f^*(w, t-1)$[3]. From $\{df^*(w, t)\}$, we derive the mean and standard deviation as $mean^*(w)$ and $stddev^*(w)$ respectively.

Table 3 shows the means and standard deviations of absolute

[3] $*$ denote RE or RT.

network	absolute energy diff.		word frequency diff.	
	$mean(\{d_t\})$	$stddev(\{d_t\})$	$mean^*(w)$	$stddev^*(w)$
G^{RE}	0.0093	0.0075	23.5	145.8
G^{RT}	0.0098	0.0093	148.3	1230.2

Table 3: Descriptive statistics of reply and retweet networks

energy difference sequence and word frequency difference sequence of the dynamic reply and retweet networks.

Take the largest difference of energy from both G^{RE} and G^{RT} on 28 June 2012 as an example. The top three words from retweets with highest frequency difference are "tax","Obamacar" and "scotu" (**S**upreme **C**ourt **o**f **U**nited **S**tates) after stopword removal and word stemming. By searching the web using these keywords, we verified that the Obamacare healthcare law was upheld by the Supreme Court of United States, and there were concerns about tax increase as its outcome. This event attracted a lot of replies and retweets on June 28. The word frequency difference of "Obamacar" in retweets subsided quickly on 29 Jun, 2012 as shown by a negative $df^{RT}(\text{"}Obamacar\text{"}, 29\ Jun)$ value.

For each day t, we define the *average frequency difference* of the three words w_1, w_2 and w_3 with highest $df^*(\cdot, t)$ as $M^*(t) = \frac{1}{3}\sum_{i=1}^{3} df^*(w_i, t)$. If $M^*(t)$ deviates far away from the mean $mean^*(w)$ w.r.t. the value $stddev^*(w)$, an event is considered to happen on day t.

Formally, we define the normalized $M^*(t)$ on day t as

$$N^*(t) = \frac{M^*(t) - mean^*(w)}{stddev^*(w)}$$

The larger the $N^*(t)$ is, the more likely the top words are able to explain some event on t. Empirically, we use the words with $N^*(t) \geq 8$ to help us to explain internal events. On the other hand, an external event may prevent people from communicating in Twitter. In this case, $N^*(t)$ may be small due to very few users generating tweets. We nevertheless tried to

date ($N^*(t)$, G^*)	event	description
5^{th} **Jun.** **(15.6,** G^{RT}**)**	wisconsin(21089) walker(20726) wirecal(16213)	Tom Barr. Wisconsin voters rejected a year-long effort to recall Gov. Scott Walker.
15^{th} **Jun.** **(10.6,** G^{RT}**)**	obama(17652) immigr(11284) illeg(10588)	President Obama is way out of line with his June 15th immigration amnesty.
20^{th} **Jun.** **(16.7,** G^{RT}**)**	fastandfuri(23295) holder(19991) obama(18974)	White House has asserted executive privilege on "fast and Furious" documents.
21^{st} **Jun.** **(2.3,** G^{RT}**)**	lebron(3816) nba(2694) twitter(2517)	Twitter goes down in worst crash in 8 months.
28^{th} **Jun.** **(36.2,** G^{RT}**)**	tax(52444) obamacar(51390) scotu(30247)	Obamacare is the largest tax increase in the history of the world.
30^{th} **Jun.** **(1.7,** G^{RE}**)**	natgat(1898) republic(1061) storm(1049)	Honorable Bio visited California to bring the power loss.
20^{th} **Jul.** **(10.0,** G^{RT}**)**	shoot(25860) gun(24103) aurora(20480)	A gunman killed 12 people at a midnight showing of the new Batman movie in Aurora, Colorado.

Table 4: Detected events from G^{RE} and G^{RT}

use the frequent words on day t to search the web to confirm if an event is external.

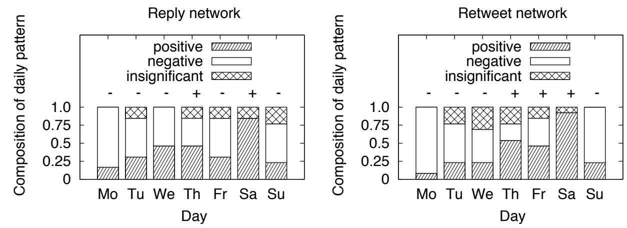
Figure 6 illustrates the $N^*(t)$ values of both G^{RE} and G^{RT}. Table 4 lists seven events found from G^{RE} and G^{RT} using $\mathcal{R}$-energy. The first column shows the date of event and $N^*(t)$ value. The second column shows the top three words derived by top frequency differences in G^{RE} or G^{RT} depending on which of the two networks is used to detect the event. The final column shows the description of events manually derived from the Google search results of the top words.

Instead of using $\mathcal{R}$-energy, we also experimented with time series of daily reply and retweet counts using a similar event detection method. Unlike the $\mathcal{R}$-energy time series, we could detect only two events on 28 June and 30 June listed in Table 4. This is due to reply and retweet counts fluctuates very much over time. We therefore detect fewer bursty events than that using $\mathcal{R}$-energy. The results also show that $\mathcal{R}$-energy can help detecting events that are different.

Periodic Trend Pattern Detection

Other than ad hoc events, Mann-Kendall trend test [23] indicates that a periodic pattern significantly exists in G^{RE} and G^{RT} of Figure 5. We also want to detect weekly trend patterns from the figure by examining the regularities in network energy changes. This weekly pattern can be even more distinct when the ad hoc events are removed.

In this section, we therefore focus on detecting weekly pattern. Based on a pre-defined threshold θ ($= 0.1 \times mean(\{d_t\})$), we first derive three kinds of energy changes from the previous day, namely (i) *energy increase* ('$+$'), (ii) *energy decrease* ('$-$'), and (iii) *insignificant change* ($null$).



(a) G^{RE} (b) G^{RT}

Figure 7: Weekly pattern detecting

Given a day of a week x, e.g., Tuesday, we count the number of '$+$'s, '$-$'s, and $null$'s and denote them by $p(x)$, $n(x)$, and $null(x)$ respectively. After ignoring the ad hoc events, we increment $p(x)$ if the energy change is more than θ; increment $n(x)$ if the energy change is smaller than $-\theta$; or increment $null(x)$ otherwise. The proportions of '$+$'s and '$-$'s on x across multiple weeks can be defined as:

$$prop(`+', x) = \frac{p(x)}{p(x) + n(x) + null(x)}$$
$$prop(`-', x) = \frac{n(x)}{p(x) + n(x) + null(x)}$$
$$prop(null, x) = \frac{null(x)}{p(x) + n(x) + null(x)}$$

Let $max_{prop}(x)$ be maximum value of $prop(`+', x)$, $prop(`-', x)$ and $prop(null, x)$. We assign a label l to day x as follows:

$$l = \begin{cases} `+', & \text{if } prop(`+', x) \text{ equal to } max_{prop}(x) \\ `-', & \text{if } prop(`-', x) \text{ equal to } max_{prop}(x) \\ null, & otherwise \end{cases} \quad (14)$$

In case of $prop(`+', x) = prop(`-', x) = max_{prop}(x)$, we assign a $null$ label to the day x.

For example, suppose out of 13 weeks, there are 12 Mondays with '$-$'s, one with '$+$' and zero with $null$. The compositions of positive, negative, and null energy changes on Monday are 7.7%, 92.3% and 0% respectively. Monday therefore is assigned to '$-$'. By assembling the proportions of positive, negative, null energy changes for different days of week, we obtain the *weekly trend pattern* of G^{RE} and G^{RT}.

Figure 7 illustrates the composition of weekly pattern for G^{RE} and G^{RT}. According to label assignment rule, we obtain the weekly trend pattern "$---+-+-$" for G^{RE}, and another weekly trend pattern "$---+++-$" for G^{RT}. Other than Friday, the two weekly trend patterns obtained from G^{RE} and G^{RT} are very similar.

From the weekly trend pattern, we can casually conclude that users are less likely to tweet on Saturdays but tweet a lot on Sundays as well as Mondays.

7. CONCLUSION

Measuring and explaining the robustness of large scale networks is an important and challenging task both in network science theories and applications. The robustness of a network is related to the connectivity of its vertices. In this paper, based on the normalized Laplacian matrix, we define a new robustness measure called $\mathcal{R}$-energy which is closely related to the average 2-step commute probability. The computation of $\mathcal{R}$-energy is highly efficient as it involves a single scan of the vertices and edges of the network. This new measure can therefore be applied to large dynamic networks. This paper also presents the results of applying $\mathcal{R}$-energy to large dynamic Twitter networks so as to detect events and trends. Our empirical study shows that interesting events and trends can be found among tweeting users.

As part of our future work, we plan to develop other robustness measures for directed and weighted networks given that such networks are common on social media (e.g., Twitter, Google+, etc.). We can also apply robustness measures to discover tightly connected sub-communities in the networks.

ACKNOWLEDGEMENTS
This work is supported by the National Research Foundation under its International Research Centre@Singapore Funding Initiative and administered by the IDM Programme Office, and National Research Foundation (NRF).

REFERENCES
1. Albert, R., Jeong, H., and Barabási, A. L. The internet's achilles' heel: Error and attack tolerance in complex networks. *Nature 406* (2000), 378–382.
2. Alvey, I. Expander graph and property. *Technique report* (2011), 1–17.
3. Balakrishnan, R. The energy of a graph. *Linear Algebra and its Applications 387* (2004), 287–295.
4. Ball, F., Mollison, D., and Scalia-Tomba, G. Epidemics with two levels of mixing. *The Annals of Applied Probability 7*, 1 (1997), 46–89.
5. Bobkov, S. G., C., H., and Tetali, P. $\lambda_{+\infty}$ vertex isoperimetry and concentration. *Combinatorica 20*, 2 (2000), 153–172.
6. Callaway, D. S., Newman, M. E. J., Strogatz, S. H., and Watts, D. J. Network robustness and fragility: Percolation on random graphs. *Physical Review Letters 85* (2000), 5468–5471.
7. Cavers, M., Fallat, S., and Kirkland, S. On the normalized laplacian energy and general randić index r_{-1} of graphs. *Linear Algebra and its Applications 433*, 1 (2010), 172–190.
8. Cho, E., Myers, S. A., and Leskovec, J. Friendship and mobility: User movement in location-based social networks (2011). 1082–1090.
9. Cohen, R., Erez, K., ben Avraham, D., and Havlin, S. Resilience of the internet to random breakdowns. *Physical Review Letters 85* (2000), 4626–4228.
10. Day, J., and So, W. Singular value inequality and graph energy change. *Electronic Journal of Linear Algebra 16* (2007), 291–299.
11. Day, J., and So, W. Graph energy change due to edge deletion. *Linear Algebra Application 428* (2008), 2070–2078.
12. Dekker, A. H., and Colbert, B. D. Network robustness and graph topology. In *ACSC* (2004), 359–368.
13. Eubank, S., Guclu, H., Kumar, V., Marathe, M., Srinivasan, A., Toroczkai, Z., and Wang, N. Modeling disease outbreaks in realistic urban social networks. *Nature 429* (2004), 180–184.
14. Fiedler, M. Algebraic connectivity of graphs. *Czechoslovak Mathematical Journal 23*, 98 (1973), 298–305.
15. Gutman, I. The energy of a graph. *BeT. Math.-Stai'ist. Sekt. FOTschnngsz. Gmz 103* (1978), 1–22.
16. Hasegawa, T., and Masuda, N. Robustness of networks against propagating attacks under vaccination strategies. *Journal of Statistical Mechanics: Theory and Experiment 429* (2011), P09014.
17. Hoory, S., Linial, N., and Wigderson, A. Expander graphs and their applications. *Journal: Bulletin of The American Mathematical Society 43*, 4 (2006), 439–562.
18. Jamakovic, A., and Mieghem, P. V. On the robustness of complex networks by using the algebraic connectivity. In *Networking* (2008), 183–194.
19. Java, A., Song, X., Finin, T., and Tseng, B. Why we twitter: Understanding microblogging usage and communities. In *WEBKDD* (2007).
20. Leskovec, J., Kleinberg, J., and Faloutsos, C. Graphs over time: Densification laws, shrinking diameters and possible explanations (2005). 177–187.
21. Leskovec, J., Lang, K., Dasgupta, A., and Mahoney, M. Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters. *Internet Mathematics 6*, 1 (2009), 29–123.
22. Malliaros, F. D., Megalooikonomou, V., and Faloutsos, C. Fast robustness estimation in large social graphs: Communities and anomaly detection. In *SDM* (2012), 942–953.
23. Mann, H. B. Nonparametric tests against trend. *Econometrica 13* (1945), 245–259.
24. Palmer, C. R., and Steffan, J. G. Generating network topologies that obey power laws. In *Globecom* (2000), 33–37.
25. Robbiano, M., and Jiménez, R. Applications of a theorem by ky fan in the theory of graph energy. *MATCH 62* (2009), 537–552.
26. Woodbury, G. An introduction to statistics. *Cengage Learning* (2001), 74.
27. Zhou, B. More on energy and laplacian energy. *MATCH 64* (2010), 75–84.
28. Zhou, B., Gutman, I., and Aleksić, T. A note on the laplacian energy of graphs. *MATCH 60* (2008), 441–446.