Complex Network Comparison Using Random Walks*

Shan Lu
Department of Electrical and Computer Engineering
University of Massachusetts
Amherst
slu@ecs.umass.edu

Jieqi Kang
Department of Electrical and Computer Engineering
University of Massachusetts
Amherst
jkang@ecs.umass.edu

Weibo Gong
Department of Electrical and Computer Engineering
University of Massachusetts
Amherst
gong@ecs.umass.edu

Don Towsley
School of Computer Science
University of Massachusetts
Amherst
towsley@cs.umass.edu

ABSTRACT

In this paper, we proposed a complex network comparison method based on mathematical theory of diffusion over manifolds using random walks over graphs. We show that our method not only distinguishes between graphs with different degree distributions, but also different graphs with same degree distributions. We compare the undirected power law graphs generated by Barabasi-Albert (B-A) model and directed power law graphs generated by Krapivsky's model to the random graphs generated by Erdos-Renyi model. We also compare power law graphs generated by four different generative models with the same degree distribution.

Keywords

random walk, complex network, power law graph, graph comparison

1. INTRODUCTION

The asymptotic behavior of the heat content has been used as a tool to understand the geometry of a manifold domain [2, 16] or the connectivity structure of a graph [12, 13]. Heat content, as the solution of the heat equation associated with the Laplacian operator, summarizes the heat diffusion in the domain as a function of time for a given initial heat distribution. One property of the heat content method is that its asymptotic behavior as $t \to 0$ separates the heat content curves of different structures. This enables one to develop fast algorithms for complex graphs comparison. In [6, 7] it was pointed out that Monte-Carlo simulations of diffusion are effective in testing the similarity of complex graphs and that such simulations provide plausible mechanisms for many brain activities. In this paper we apply the random walk method to distinguish between complex networks, with experiments on graph comparison between graphs with different distributions and the graphs with the same distributions but different connectivity structures.

Graph comparison is a challenging task since graph sizes increase extremely fast in diverse areas, such as social networks (Facebook, Twitter), Web graphs (Google), knowledge networks (Wikipedia), etc.. Many graph comparison methods have been proposed to quantitatively define the similarity between graphs. In [10] the authors summarize the existing methods into three categories: graph isomorphism, iterative methods and feature extraction. The graph isomorphism and iterative methods are not scalable and thus not effective for large networks. Feature extraction methods extract features like degree distribution, eigenvalues to compare. These methods are closer in spirit to our method. However previously proposed features may not reflect the network connectivity structure very well. For example, in [13], the authors give an example where two isospectral nonisometric planar graphs can be distinguished by the heat content, despite the fact they share the same set of eigenvalues. In [8], the authors analysed the structural properties of graphs with the same degree distribution and found that different networks with the same degree distribution may have distinct structural properties. In [15], the authors discussed methods for similarity testing in directed web graphs, including vertex ranking, sequence similarity and signature similarity, among others. However, like most of the algorithms in [10], they need to know the node correspondence, which is the mapping between the graphs' nodes. It is already a hard problem for many complex networks.

Our algorithm exhibits the following features. First, our method summarizes graph structure into a single time function so as to facilitate similarity testing. Second, the behavior of this function around time $t = 0$ is the most important for the comparison. Practically we only need the beginning part of the heat content so that we can greatly reduce the computation time. Third, we use lazy random walk to estimate the heat content function, thereby avoid computing the

*This work is supported in part by the United States National Science Foundation Grant EFRI-0735974, CNS-1065133 and CNS-1239102, and Army Research Office Contract W911NF-08-1-0233 and W911NF-12-1-03.

eigenvalues and eigenvectors of the graph Laplacian while retaining the spectral information. Fourth, our algorithm only compares the connectivity structure and does not use node correspondence. Hence it avoids the need to identify a mapping between the graphs' nodes. Finally we note that our method is robust to minor changes in large graphs according to the interlacing theorem in [3]. With these features, our algorithm is capable of handling very large complex networks. Using experiments, we show that our algorithm performs better in distinguishing networks comparing to the other feature extraction methods, such as eigenvalues and degree distributions.

The rest of the paper is organized as follows. In Section 2, we give the notations and review the concept of heat equation and heat content for graphs. In Section 3, we use the lazy random walk simulation method to estimate the heat content. In Section 4, the graph generative models used in experiment part are introduced. Experiment settings and results are presented in Section 5. Section 6 summarizes the main results and discusses future work.

2. HEAT EQUATION AND HEAT CONTENT

2.1 Notations

Let $G = (V, E)$ denote a graph with vertex set V and edge set $E \subseteq V \times V$ with adjacency matrix $A = [a_{uv}]$. $a_{uv} = 1$ if there is an edge from u to v; otherwise $a_{uv} = 0$. The out-degree matrix $D = diag[d_u]$ with $d_u = \sum_v a_{uv}$.

The graph Laplacian of a graph is defined as $L = D - A$ and the normalized Laplacian is defined as [4]

$$\mathcal{L} = D^{-1/2} L D^{-1/2}.$$

With the random walk Laplacian $L_r = D^{-1}L$, we have the following relation between $\mathcal{L}$ and L_r

$$L_r = D^{-1/2} \mathcal{L} D^{1/2}.$$

Without loss of generality, we assume that the Laplacian matrix L is diagonalizable and hence $\mathcal{L}$ is diagonalizable. Let $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$ be the eigenvalues of $\mathcal{L}$ and $\phi_i, i = 1, \cdots, n$ be the corresponding eigenvectors. With $\Lambda = diag[\lambda_i]$ and $\Phi = [\phi_1, \cdots, \phi_n]$ we can diagonalize $\mathcal{L}$ as

$$\mathcal{L} = \Phi \Lambda \Phi^{-1},$$

where $\Phi^{-1} = [\pi_1; \pi_2; \cdots; \pi_n]$.

Furthermore we have

$$L_r = (D^{-1/2}\Phi)\Lambda(D^{-1/2}\Phi)^{-1}. \quad (1)$$

L_r and $\mathcal{L}$ share the same set of eigenvalues, but the corresponding eigenvectors are different. $\mathcal{L}$ is the normalized graph Laplacian used in the heat equation on a graph. We use the relationship between $\mathcal{L}$ and L_r to develop a random walk simulation method in the later section.

2.2 Heat equation and heat content

Vertex set V is partitioned into two subsets, the set of all interior nodes iD and the set of all boundary nodes ∂D. We have $V = iD \cup \partial D$. The heat equation associated with the normalized graph Laplacian is

$$\begin{cases} \frac{\partial H_t}{\partial t} = -\mathcal{L} H_t \\ H_t(u, v) = 0 \text{ for } u \in \partial D, \end{cases} \quad (2)$$

with initial condition

$$H_0(u, u) = \begin{cases} 1 & \text{if } u \in iD \\ 0 & \text{else .} \end{cases}$$

Assuming the total number of vertices is N and the number of interior vertices is n, H_t is an $N \times N$ matrix. $H_t(u, v)$ measures the amount of heat that flows from vertex u to vertex v at time t. All heat that flows to the boundary vertices is absorbed. We label the interior vertices $1, \cdots, n$ and the boundary vertices $n+1, \cdots, N$. The normalized Laplacian $\mathcal{L}$ can be partitioned into four parts

$$\mathcal{L} = \begin{bmatrix} \mathcal{L}_{iD,iD} & \mathcal{L}_{\partial D,iD} \\ \mathcal{L}_{iD,\partial D} & \mathcal{L}_{\partial D,\partial D} \end{bmatrix}.$$

Since we are only interested in the heat remaining in the interior domain, define the $n \times n$ matrix h_t with

$$h_t(u, v) = H_t(u, v) \text{ for } u, v \in iD.$$

The solution to the heat equation is $h_t = e^{-\mathcal{L}_{iD,iD} t}$. For convenience, we slightly abuse notation and still use Λ and Φ as the eigenvalue matrix and eigenvector matrix of $\mathcal{L}_{iD,iD}$. We have

$$h_t = \Phi e^{-\Lambda t} \Phi^{-1}.$$

For each entry of h_t, we have

$$h_t(u, v) = \sum_{i=1}^{n} e^{-\lambda_i t} \phi_i(u) \pi_i(v). \quad (3)$$

The heat content $Q(t)$ is defined as the sum of all the entries in h_t:

$$Q(t) = \sum_u \sum_v h_t(u, v) \quad (4)$$

$$= \sum_u \sum_v \sum_{i=1}^{n} e^{-\lambda_i t} \phi_i(u) \pi_i(v). \quad (5)$$

Letting $\alpha_i = \sum_u \sum_v \phi_i(u)\pi_i(v)$ yields

$$Q(t) = \sum_{i=1}^{n} \alpha_i e^{-\lambda_i t}. \quad (6)$$

The heat content can be viewed as the sum of exponentially decaying functions with different rates and different strengths. The rates and strengths are determined by the graph Laplacian eigenvalues and eigenvectors, respectively. To emphasize the larger eigenvalues more, we can use the following derivatives of the heat content for comparison:

$$\dot{Q}(t) = -\sum_{i=1}^{m} \alpha_i \lambda_i e^{-\lambda_i t},$$

3. RANDOM WALK METHODS FOR HEAT CONTENT ESTIMATION

Computing eigenvalues and eigenvectors of the Laplacian matrix needed for evaluating the heat content is very time consuming for large complex networks. In this section, we use a discrete time random walk method to approximate the heat content.

We consider a random walk where the random walker moves from vertex u to a neighboring vertex v with probability

a_{uv}/d_u. Define the transition matrix $M = D^{-1}A$ and the lazy random walk transition matrix as

$$M_L = (1-\delta)I + \delta M.$$

In other words, a random walker either moves to one of the neighboring vertex with probability δ or remains at the current vertex with probability $1-\delta$. For any given time $t = k\delta$, we have

$$P_t = M_L^k P_0 = [I - \frac{t}{k}L_r]^k P_0 \ \rightarrow \ e^{-L_r t}P_0. \qquad (7)$$

Here the arrow ($\rightarrow$) implies taking the limit with $k \rightarrow \infty$ (at the same time $\delta \rightarrow 0$ while keeping $k\delta = t$). P_0 is the initial distribution of a random walker. We have $M_L^k \rightarrow e^{-L_r t}$. It can be seen from equation (1) that each entry in matrix M_L^k converges to

$$M_L^k(u,v) \rightarrow \sum_{i=1}^{n} e^{-\lambda_i t}\phi_i(u)\pi_i(v)\sqrt{\frac{d_v}{d_u}}.$$

Comparing to equation (3), we have

$$M_L^k(u,v)\sqrt{\frac{d_u}{d_v}} \rightarrow h_t(u,v).$$

$M_L^k(u,v)$ measures the probability that a random walker starting at vertex u ends up at vertex v in k steps in the lazy random walk. Now, with equation (5), we obtain the approximation for $Q(t)$:

$$\hat{Q}(t) = \sum_{u \in iD}\sum_{v \in iD} M_L^k(u,v)\sqrt{\frac{d_u}{d_v}}. \qquad (8)$$

With the lazy random walk approximation, our algorithm avoids the computation of the eigenvalues and the eigenvectors. Instead of computing $M_L^k(u,v)$ with matrix multiplications, we can use the Monte Carlo method to estimate $M_L^k(u,v)$. In the Monte-Carlo simulation, we start with the same number of random walkers on every vertex. By the law of large numbers (LLN), the variance of the estimated value is inversely proportional to the amount of random walkers. Therefore, our method provides a trade off between precision and computation time. In fact, we find that a small number of the random walkers at each vertex works well on large graphs with a large number of vertices.

4. GENERATIVE MODELS

We consider the following generative models for complex graphs: (1) Barabasi-Albert model [1], which generates undirected graphs with power law degree distributions; (2) Erdos-Renyi model, which generates random graphs with binomial degree distribution. (3) Krapivsky's model[11], which generates directed graphs with bi-variate power law degree distributions. (4) Four models to generate random graphs with a given distribution. They are Molloy-Reed model; Kalisky model; model A and model B [8].

4.1 Generative models for undirected random graphs

Erdos-Renyi (E-R) model

The graph is constructed by connecting nodes randomly and independently. An edge is added to each pair of vertices with a given probability.

Barabasi-Albert (B-A) model

The construction starts with m_0 initial nodes. Each new node is connected to $m (m \leq m_0)$ existing nodes with a probability proportional to the number of links that the existing nodes already have. The degree distribution follows $P(D = d) \sim d^{-3}$.

In the experiment section, we will compare the graphs with power law degree distributions generated by the B-A model to the E-R random graphs. We will also compare the graphs generated by the same generative model (B-A model) with different parameters.

4.2 Generative models for directed graphs

In [11], the authors proposed a graph generative model to describe growing processes in the Web Graphs (WG). This model includes two separated processes: (1) with probability p, a new node is introduced and immediately attaches to an existing node u with probability proportional to $d_u^{\text{in}} + \lambda_{\text{in}}$, where d_u^{in} is the in-degree of node u; (2) with probability q, a new edge is created. From existing node v to node u, a new edge is created with probability proportional to $(d_u^{\text{in}} + \lambda_{\text{in}})(d_v^{\text{out}} + \lambda_{\text{out}})$, where d_v^{out} is the out-degree of node v. This model produces directed graphs with marginal in-degree and out-degree distributions that are both heavy tailed. Let $P(d^{\text{in}} = i) \sim i^{-v_{\text{in}}}$ and $P(d^{\text{out}} = j) \sim j^{-v_{\text{out}}}$. We have $v_{\text{in}} = 2 + p\lambda_{\text{in}}$ and $v_{\text{out}} = 1 + q^{-1} + p\lambda_{\text{out}}/q$. The average in-degree and out-degree both equals to $1/p$.

In the experiment part, we will apply our method to compare the directed power law graphs generated by the 'WG' model to the directed graphs generated by the E-R model. Directed E-R graphs are generated by independently adding directed edges from a node to a target node with a given probability.

4.3 Generative models for power law graphs with a given degree distribution

In [8], the authors used the following four models to generate different networks with the same degree distribution.

Molloy-Reed Model (M-R Model) [14]

Assign a degree to each vertex. Randomly connect a pair of vertices and select each vertex with probability proportional to its number of open connections.

Kalisky Model [9]

Assign a degree to each vertex. Start from the vertex with the maximal degree and exhaust its open connections by randomly connecting it to other vertices. These vertices are the first layer vertices. The second layer vertices are selected by randomly connecting the remaining open connections in the first layer. Repeat until there is no open connection.

Model A

Assign a degree to each vertex. Randomly connect maximal degree node to the available vertices. Repeat the procedure until there is no open connection.

Model B

Model B is the same as model A, except that the vertices connecting to the maximal degree node are selected in sequence according to a given vertices list.

Both model A and model B are new methods proposed in [8]. In our experiments, we will reuse the four models in [8] to generate groups of networks to compare. We will show that our feature, heat content, is better in representing the network structure comparing to the degree distribution.

5. EXPERIMENT RESULTS

We first test our method for comparing between graphs with different degree distributions but similar Laplacian spectra. In the last experiment, we illustrate the experiment results on graphs with the same degree distribution to show our method's ability in detecting graphs' structural differences.

5.1 Undirected Graphs with different degree distributions (B-A model vs. E-R model)

Two groups of graphs are generated using the B-A model and E-R model respectively. The total number of nodes is 2000. Each group includes four graphs with average degree varying from 20 to 50. Boundary vertices are defined to be the 40 vertices with the smallest degrees. The approximated heat contents $\hat{Q}(t)$ for the 8 graphs are plotted in Fig. 1(a).

As shown in the figure, the heat contents of the two groups of graphs follow different patterns. When t is close to zero, the heat contents for power law graphs drops faster than for E-R random graphs, but the decrease speed slows down once $t > 5$. On the other hand, the decrease rate for heat contents associated with E-R random graphs is comparatively more constant throughout the process. The difference between the heat contents of these two types of graphs is illustrated more clearly if we focus on the time derivative of the heat content, as shown in Fig. 1(b). When we compare the heat content derivatives for the power law graphs, the derivatives at the beginning part are in the order of the average degrees (as shown in Figure 2(b)). Using the heat content method, we can also differentiate graphs with different mean degrees generated from the same B-A model.

Spectrum of the Laplacian: For the spectra of these two kinds of graphs, Chung *et.al.*[5] proved that eigenvalues of the normalized Laplacian satisfy the semicircle law under the condition that the minimum expected degree is relatively large. Both E-R random graphs and power law graphs satisfy this condition as indicated in [5]. Meanwhile, the paper also proves that if two graphs have the same mean degree, the circle radius will be almost the same (as shown in Fig. 3).

We observe that using only the Laplacian spectrum we can hardly distinguish the two types of graphs. However, according to equation (6), the values of α_i also play an important role in the heat contents. In Fig. 4, the Laplacian eigenvalues (except the smallest one) and corresponding α values are plotted with the x-axis as the index. The eigenvalues of the two graphs, which are plotted in dashed line, are too similar to compare. But the strengths (α) for the power law graph are much larger than those for the E-R random graph. With the larger α values, the effect of the larger eigenvalues on the heat content are highlighted. The heat content for the power law graphs decreases faster at the beginning part in Fig. 1(a). For E-R random graphs, which are more homogeneous in terms of graph structure, the larger eigenvalues

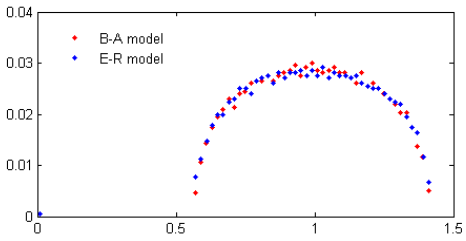


Figure 3: The Laplacian spectrum distribution of one power law graph and one random graph with mean degree 20

in the Laplacian spectrum have much smaller weights, thus do not impact the heat content behavior much.

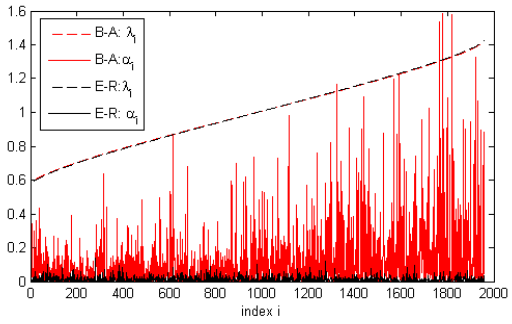


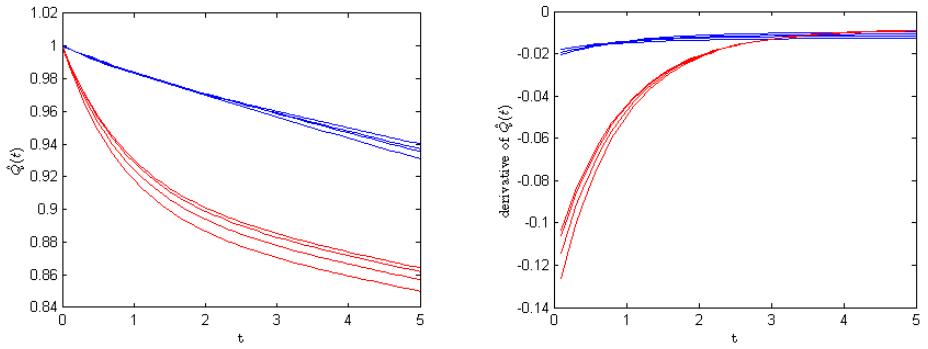
Figure 4: Eigenvalues and corresponding α values

Our analysis shows that the eigenvalues of the Laplacian alone cannot distinguish complex networks. The eigenvectors are obviously needed although computing both the eigenvalues and eigenvectors is itself a big obstacle in large scale networks. However, our algorithm can avoid this difficulty by estimating the heat content of the graphs using random walks on graph.

5.2 Directed Graphs with different degree distributions (Krapivsky's model vs. E-R model)

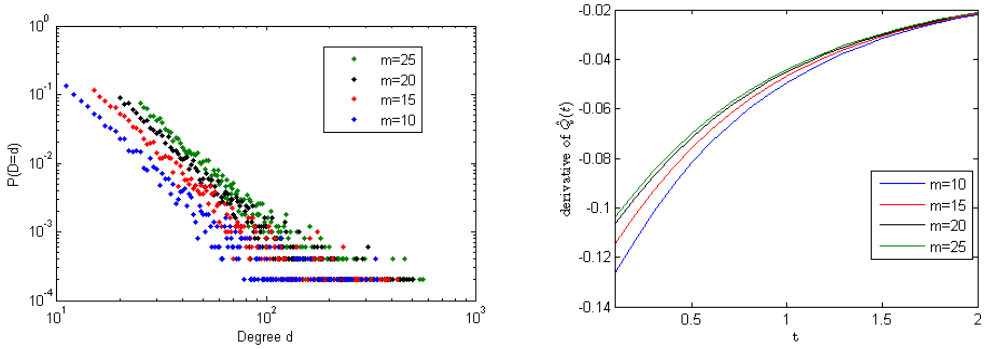
Two groups of directed graphs are generated using Krapivsky's 'WG' model and the E-R model respectively. The total number of nodes is 2000. Each group contains four graphs with different average degrees. We change the average degrees of directed power law graphs by setting p to be 0.1, 0.15, 0.2 and 0.25. Boundary vertices are defined to be the 200 vertices with the smallest in-degree out-degree products (vertices with zero in-degrees are not candidates for boundaries). The approximated heat contents $\hat{Q}(t)$ and derivatives are plotted in Fig. 5.

As shown in the figure, the directed power law graphs and E-R random graphs exhibit similar behavior to undirected graphs. The two groups of graphs can be separated immediately by comparing the heat contents.



(a) Heat Content (b) Heat Content Derivatives

Figure 1: Undirected graph comparison (red: power law graphs; blue: E-R random graphs)



(a) Degree Distributions (b) Heat Content Derivatives

Figure 2: Degree distributions and heat content derivatives for power law graphs with different mean degree

5.3 Graphs with the same degree distribution

In this section, we test our algorithm's ability in distinguishing graphs with the same degree distribution. We first generate a 2000 nodes power law graph using B-A model with $m = 2$. Then we generate 3 graphs for each 4 generative models (Molloy-Reed model, Kalisky model, model A and model B) with the same total number of nodes and degree distribution. The heat contents and derivatives of all the 13 graphs are shown in Figure 6(a) and 6(b).

As we can see from the results, graphs with the same degree distribution can still be well distinguished according to their heat contents behaviors. The heat contents for graphs generated by the same model are clustered. And at the same time, although with the same degree distributions, the differences of the heat contents between the 5 generative models are also noticeable. We also notice that the heat contents for model B (the yellow curves) perform much differently from the other four models (Molloy-Reed model, Kalisky model, model A and the B-A model). This result is consistent with the conclusion in [8] that, model B gives decentralized and low efficient network, while the others are more centralized and high efficient.

6. CONCLUSION

In this paper, we proposed a random walk method to estimate the heat content on graphs. We first apply the method to compare graphs with different degree distributions. Graphs with heavy tailed degree distribution have different heat content curves comparing to the random graphs generated by the E-R model: the decrease rate for the previous is much larger than that for the later at the very beginning part. Our method can also distinguish graphs with the same degree distribution but different structural properties. Experiments show that, our algorithm is better in graph comparison than some other feature extraction methods like eigenvalues and degree distributions. In our future work, we will apply our approach to more general problems in graph comparison. For example, we will use the method on graphs other than those generated by E-R and B-A models. We will also consider real world network datasets.

7. REFERENCES

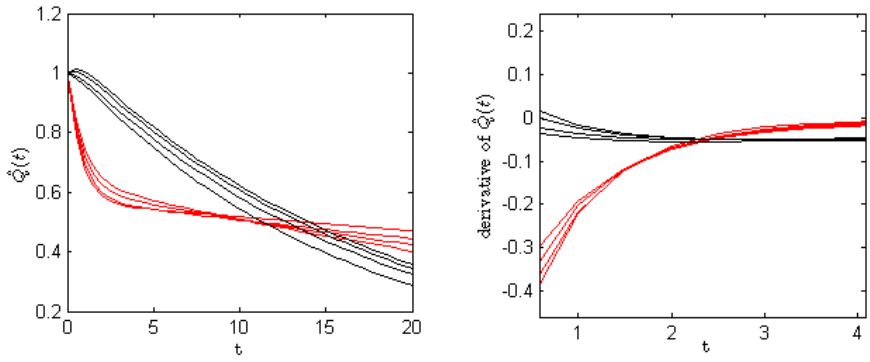
[1] A. L. Barbasi and R. Albert. Emergence of scalin in random networks. *Science*, 286:509–512, 1999.

[2] M. V. D. Berg and P. B. Gilkey. Heat content asymptotics of a riemannian manifold with boundary. *Journal of Function Analysis*, 120:48–71, 1994.

[3] S. Butler. Interlacing for weighted graphs using the normalized laplacian. *Electronic Journal of Linear Algebra*, 16:90–98, 2007.

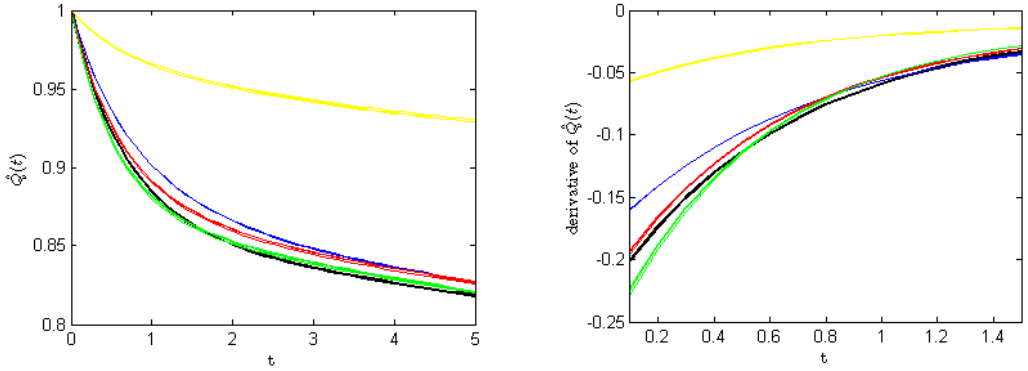
[4] F. Chung. *Spectral Graph Theory*. American Mathematical Society, 1997.

[5] F. Chung, L. Lu, and V. Vu. Spectra of random



(a) Heat content (b) Heat content derivative

Figure 5: Directed graph comparison (red line: power law graphs; black line: E-R random graphs)



(a) Heat content (b) Heat content derivative

Figure 6: Comparing graphs with the same degree distribution (Black: B-A; Blue: M-R; Red:Kalisky; Green: model A; Yellow: model B)

graphs with given expected degrees. *Proceedings of the National Academy of Sciences of the United States of Amherica*, 100(11):6313–6318, 2003.

[6] W. Gong. Can one hear the shape of a concept? In *Proceedings of the 31st Chinese Control Conference (Plenary Lecture)*, pages 22–26, Hefei, China, July 2012.

[7] W. Gong. Transient response functions for graph structure addressable memory. to appear. In *Proceedings of the 52th IEEE Conference on Decision and Control*, Florence, Italy, December 2013.

[8] J.H.H.Grisi-Filho, R. Ossada, F. Ferreira, and M. Amaku. Scale-free networks with the same degree distribution: Different structural properties. *Physics Research International*, 2013:1–9, 2013.

[9] T. Kalisky, R. Cohen, D. Ben-Avraham, and S. Havlin. Tomography and stability of complex networks. *Lecture Notes in Physics*, 650:3–34, 2004.

[10] D. Koutra, A. Parikh, A. Ramdas, and J. Xiang. Algorithms for graph similarity and subgraph matching. *http://www.cs.cmu.edu/ ./aram-das/reports/DBreport.pdf*, 2011.

[11] P. L. Krapivsky, G. J. Rodgers, and S. Redner. Degree distributions of growing networks. *Physical Review Letters*, 86:5401–5404, 2001.

[12] P. Mcdonald and R. Meyers. Diffusion on graphs, poisson problems and spectral geometry. *Transaction of the American Mathematical Society*, 354(12):5111–5136, 2002.

[13] P. Mcdonald and R. Meyers. Isospectral polygons, planar graphs and heat content. *Proceedings of the American Mathematical Society*, 131(11):3589–3599, 2003.

[14] M. Molly and B. Reed. The size of the giant component of a random graph with a given degree sequence. *Combinatorics, Probability and Computing*, 7:295–305, 1998.

[15] P. papadimitriou, A. Dasdan, and H. Carcia-Molina. Web graph similarity for anomaly detection. *Journal of International Service and Applications*, 1(1):19–30, 2010.

[16] J. Park and K. Kim. The heat energy content of a riemannian manifold. *Trends in Mathematics, Information Center for Mathematical Sciences*, 5(2):125–129, 2002.