Published in "Chaos: An Interdisciplinary Journal of Nonlinear Science
23(1): 013104–013104–7, 2013"
which should be cited to refer to this work.

Spectral coarse graining for random walks in bipartite networks

Yang Wang,[1,2] An Zeng,[3,a)] Zengru Di,[1,2] and Ying Fan[1,2,b)]
[1]*Department of Systems Science, School of Management, Beijing Normal University, Beijing 100875, People's Republic of China*
[2]*Center for Complexity Research, Beijing Normal University, Beijing 100875, People's Republic of China*
[3]*Department of Physics, University of Fribourg, Chemin du Musée 3, CH-1700 Fribourg, Switzerland*

Many real-world networks display a natural bipartite structure, yet analyzing and visualizing large bipartite networks is one of the open challenges in complex network research. A practical approach to this problem would be to reduce the complexity of the bipartite system while at the same time preserve its functionality. However, we find that existing coarse graining methods for monopartite networks usually fail for bipartite networks. In this paper, we use spectral analysis to design a coarse graining scheme specific for bipartite networks, which keeps their random walk properties unchanged. Numerical analysis on both artificial and real-world networks indicates that our coarse graining can better preserve most of the relevant spectral properties of the network. We validate our coarse graining method by directly comparing the mean first passage time of the walker in the original network and the reduced one.

Bipartite networks are naturally suited to understanding and modeling many real systems. However, when the network contains a very large number of nodes, it becomes practically impossible to deal with dynamical processes on it. A promising way to address this problem is to coarse grain the network, namely, to reduce the networks' complexity by mapping the large network into a smaller one while keeping its dynamical properties unchanged. Unlike monopartite networks which only contain one kind of nodes, bipartite networks consist of two distinct sets of nodes, such that links cannot exist between nodes in the same set. The dynamics on both types of nodes should be preserved in coarse graining. Additionally, the coarse graining should preserve the bipartite structure of the network. However, existing coarse graining methods for monopartite networks cannot achieve these two objectives. In this paper, we propose a spectral coarse graining method to preserve the random walk properties for bipartite networks. By introducing for each set of nodes a stochastic matrix, our method treats the two different kinds of nodes separately. As a result, the reduced networks remain bipartite. Both artificial and real bipartite networks are considered, and we find that the reduced networks have very similar spectral properties to the original ones. We validate our method by comparing the mean first passage time in the original and reduced networks. Our method can be easily extended to preserve many other spectral-determined dynamical properties in bipartite networks.

I. INTRODUCTION

As an effective way to model many real systems, complex networks have been intensively studied in the past decade. Examples range from social relationships among individuals, to interactions of proteins in biological systems, to the interdependence of function calls in large software projects. Network analysis has greatly helped us understand the structure and function of real-world systems.[1–6]

Bipartite networks, an important kind of complex network, are composed of two types of nodes with no links connecting nodes of the same type. For example, the e-commercial systems consisting of online users and products,[7,8] the scientific collaboration system consisting of authors and papers,[9,10] and family name inheritance system consisting of babies and names[11] are naturally described by such networks. So far, some topological properties such as clustering coefficient and modularity of bipartite networks have been studied.[12–14] However, one of the most difficult hurdles in analyzing and visualizing bipartite network is the size of real-world systems. The online commercial systems, for instance, can have thousands of products and even millions of users. Given that most of the algorithms used to extract the properties of the bipartite network run in times that grow polynomially with the system size, dealing with systems with very large size becomes a challenge.

In order to solve the problem mentioned above, a promising way is to consider some units of the system as almost indistinguishable and to merge them into one, i.e., to reduce the number of nodes and edges by mapping the network with N nodes and E edges into a smaller one with $\bar{N}$ nodes and $\bar{E}$ edges. Based on this concept, several coarse graining schemes for monopartite network including k-core decomposition,[15,16] box-covering process,[17,18] geographical coarse graining,[19] spectral coarse graining[20,21] have been proposed.

Specifically, the k-core decomposition intends to classify nodes into different shells which represent their importance. This technique can be used to identify the central core of a network, and was also shown to be extremely effective for

a)an.zeng@unifr.ch.
b)yfan@bnu.edu.cn.

visualization purposes. The box-covering technique yields a new network which can preserve some of the topological features of the original one. The geographical coarse graining uses a renormalization-group like numerical analysis to reduce the size of the network while preserving the degree distribution, clustering coefficient, and assortativity correlation. Spectral coarse graining methods, on the other hand, focus on the dynamical processes taking place on networks. They merge nodes based on the eigenvectors of different matrices, so that some spectral-determined dynamical processes such as random walk and synchronization on the original network are kept unchanged. Mathematically, the spectral-based methods consist in preserving some eigenvalues of the stochastic matrix or the graph Laplacian. In addition, some works have been dedicated to coarse grain networks for dynamics of heterogeneous oscillators[22] and other critical phenomena.[23]

A problem very close related to coarse graining is the community detection (CD), which groups nodes based on the link density. Because of the importance and the complexity of finding meaningful communities, recent years have witnessed an explosion of research on community structure in graphs, and a very large number of methods and techniques have been designed[24–30] (see, Ref. 31, for a review). However, there is often no clear statement on which properties of the initial network are preserved in the network of clusters.

Though the coarse graining methods mentioned above perform well in monopartite networks, they usually face problems when directly extended to directed or bipartite networks. In directed networks, the role of nodes in dynamics cannot be well characterized by the eigenvectors since imaginary eigenvalues emerge when the adjacency and Laplacian matrix are asymmetric. This problem is solved by using the paths to determine the similarity between nodes and finally preserve the dynamical properties (synchronization) when merging nodes.[32] For bipartite networks, the situation can be even more complicated. There are two types of nodes in bipartite networks and the dynamics on both types of nodes should be preserved. More importantly, the coarse graining method should preserve the intrinsic bipartite structure of the networks (i.e., no link exists between nodes of the same type). However, if we regard the bipartite networks as monopartite ones and directly apply the existing coarse graining methods, nodes from different sets will be merged. Furthermore, using the community detection methods to coarse grain bipartite networks may significantly change the network function.[13,33] As a result, it is still a challenge to preserve both the network function and the bipartite structure in coarse graining.

In this paper, we introduce a spectral-based approach to coarse grain bipartite networks. Unlike coarse graining methods for monopartite networks, our goal is to obtain a reduced bipartite network that preserves both the random walk properties of the original network and the bipartite structure. In order to preserve the random walk properties of both types of nodes, two matrices (denoted by $\mathbf{W}_m$ and $\mathbf{W}_n$) based on the stochastic matrices of the bipartite network are introduced and a new coarse graining scheme is designed. The obtained network remains bipartite and several largest nontrivial eigenvalues of $\mathbf{W}_m$ and $\mathbf{W}_n$ are preserved. Moreover, we validate our method by performing a direct test of the mean first passage time (MFPT) of random walkers on artificial and real-world bipartite networks. The new method is robust in various kinds of bipartite networks and the choices of sinks. Finally, we remark that this method can be easily extended to preserve many other spectral-determined dynamical properties in bipartite networks.

II. SPECTRAL COARSE GRAINING METHOD ON BIPARTITE NETWORKS

A. Random walks on monopartite networks

Random walks play a central role in dynamical properties taking place on complex networks.[34] Starting at some specific initial vertices, the walker jumps with equal probability from its current location to one of its nearest neighbors at each time step. A monopartite network $G = (V, E)$ with N nodes and E link can be described by the adjacency matrix $\mathbf{A}$ with elements $A_{ij} = 1$, if there is an edge connecting vertices i and j, otherwise 0. Let $p_i(t)$ be the probability that the walker is at vertex i at time step t. If the walker is at vertex j at time step $t-1$, the probability of taking a jump along any of its neighbors is $1/k_j$. Accordingly, $p_i(t)$ on an undirected monopartite network is given by

$$p_i(t) = \sum_j \frac{A_{ij}}{k_j} p_j(t-1), \tag{1}$$

where k_j is the degree of vertex j. As a matrix form, Eq. (1) can be written as $\vec{p}(t) = \mathbf{A}\mathbf{D}^{-1}\vec{p}(t-1)$, where $\vec{p}$ is the vector with elements p_i and $\mathbf{D}$ is the diagonal matrix with the degrees of the vertices down its diagonal $\mathbf{D} = diag(d_1, d_2, ..., d_N)$. Defining a stochastic matrix $\mathbf{W} = \mathbf{D}^{-1}\mathbf{A}$, random walk in monopartite network can be characterized by the stochastic matrix $\mathbf{W}$, and the element w_{ij} describes the probability that a walker goes from node i to node j.

The MFPT is an important characteristic of random walks.[34,39] To compute it exactly, one usually considers some nodes as traps. The normalized Laplacian matrix of the network is defined as $\mathbf{L} = \mathbf{I} - \mathbf{D}^{-1}\mathbf{A}$, where $\mathbf{I}$ is the identity matrix. We use Γ to denote the set of traps and $|\Gamma|$ to represent the number of traps. For simplicity, we distinguish all nodes in the network by assigning each of them a unique number. We label consecutively all nodes, excluding those in Γ, from 1 to $N - \Gamma$ and sinks are labeled from $N - \Gamma + 1$ to N. By suppressing the last $|\Gamma|$ rows and columns of the normalized Laplacian matrix, we obtain a submatrix of the normalized Laplacian matrix $\mathbf{L}$ as $\mathbf{L}'$.

The first passage time T_i is defined as a particle first arriving at any one of the traps given that it starts from node i. It is shown in Ref. 40 that the first passage time can be expressed as

$$T_i = \sum_{j=1}^{N-|\Gamma|} l_{ij}^{-1}, \tag{2}$$

where l_{ij}^{-1} is the elements of matrix $\mathbf{L}'$. Then the MFPT $\langle T \rangle$, which is defined as the average of T_i over all randomly chosen initial nodes (excluding traps), is given by

$$\langle T\rangle = \frac{1}{N}\sum_{i=1}^{N-|\Gamma|} T_i = \frac{1}{N}\sum_{i=1}^{N-|\Gamma|}\sum_{j=1}^{N-|\Gamma|} l_{ij}^{-1}. \quad (3)$$

Eq. (3) can also be found in the literature in several equivalent forms.[35,36]

Thus, the exact solution of the MFPT of the unbiased random walk is given, independently from the number and the location of the sinks. Equations (2) and (3) can reduce the problem of computing the MFPT to calculating the inverse matrix $\mathbf{L}'$ and also can be used to check the MFPT of different networks in Sec. III B.

B. Random walks on bipartite networks

In bipartite networks, connections between vertices are also described by the adjacency matrix. Different from the monopartite network, a bipartite network consists of two sets of non-overlapping nodes, and links can only exist between two nodes from distinct sets. The adjacency matrix $\mathbf{A}$ of a bipartite network is with order $M \times N$, where M and N are the number of vertices in these two distinct sets. In this paper, we call these two types of nodes as top and bottom nodes, respectively. If there is a link between vertices i in the top set and j in the bottom set, the element $A_{ij} = 1$, otherwise $A_{ij} = 0$. In bipartite networks, the random walk process is closely related to the information filtering algorithms.[37,38] Unlike monopartite networks, there are two stochastic matrices for random walk in bipartite networks. If a walker goes from the top set to the bottom set, the process is described by the stochastic matrix $\mathbf{U}$ with order $M \times N$. In $\mathbf{U}$, the element $U_{ij} = A_{ij}/k_i$. If the walker is from the bottom set to the top set, then the stochastic matrix $\mathbf{V}$ is with order $N \times M$ and element $V_{ij} = A_{ji}/k_i$. $\mathbf{U}$ and $\mathbf{V}$ contain all the information of random walk in a bipartite network.

Furthermore, we define two new matrices $\mathbf{W}_m$ and $\mathbf{W}_n$ as: $\mathbf{W}_m = \mathbf{U} \times \mathbf{V}$ and $\mathbf{W}_n = \mathbf{V} \times \mathbf{U}$. Just like the stochastic matrix in monopartite networks, $\mathbf{W}_m$ and $\mathbf{W}_n$ are square matrices. $\mathbf{W}_m(\mathbf{W}_n)$ describes the random walkers going from top (bottom) nodes to top (bottom) nodes. These two matrices have some interesting properties. In particular, the largest eigenvalue of these two matrices is equal to 1 and the elements of the corresponding eigenvector are equal. Moreover, there are several largest eigenvalues of these two matrices with the same value. As discussed in Ref. 21, eigenvectors corresponding to the eigenvalues close to 1 of the stochastic matrix $\mathbf{W}$ capture the large-scale behavior of the random walk in monopartite networks. The fact is also true in $\mathbf{W}_m$ and $\mathbf{W}_n$ in bipartite networks since they are square matrices just like $\mathbf{W}$. Therefore, our goal is to preserve the largest nontrivial eigenvalues of $\mathbf{W}_m$ and $\mathbf{W}_n$. In this way, we can preserve the properties of random walk in bipartite networks.

C. Spectral coarse graining method for bipartite networks

Now we describe the new coarse graining method. We denote the eigenvalues of a matrix $\mathbf{W}_m$ or $\mathbf{W}_n$ as λ_α and their corresponding eigenvectors $\vec{p}_\alpha$. First of all, two nodes i and j with exactly the same neighbors should be merged since they cannot be distinguished from the point of view of random walk. In the eigenvector $\vec{p}_\alpha$ for any $\lambda_\alpha \neq 0$ of $\mathbf{W}_m$ or $\mathbf{W}_n$, $p_\alpha^i = p_\alpha^j$. After merging, the new node will carry all the edges of nodes i and j and the resulting adjacency matrix of a bipartite network $\tilde{A}$ will have order $(M-1)\times N$ or $M \times (N-1)$, with the corresponding row or column of the new node being the sum of the row (column) i and j. The properties of random walk in the new bipartite network are exactly the same as those in the original network. Moreover, if $p_\alpha^i \approx p_\alpha^j$ we could also group them in order to obtain an even smaller bipartite network. By definition, if $|p_\alpha^i - p_\alpha^j| \propto \epsilon$ we could group node i and j together. Like Refs. 20 and 21, the condition $|p_\alpha^i - p_\alpha^j| \propto \epsilon$ can be implemented by defining a parameter I as the number of equally distributed intervals between the minimum and the maximum components of each eigenvector $\vec{p}$. The nodes whose eigenvector components in $\vec{p}$ fall in the same interval should be grouped.

We summarize the bipartite network spectral coarse graining (BSCG) method in the following procedures:

1. For any given bipartite network $\mathbf{A}$, we can get two stochastic matrices $\mathbf{U}$ and $\mathbf{V}$ which gives the transition probability from the top nodes to bottom nodes and bottoms nodes to top nodes, respectively;
2. Based on $\mathbf{U}$ and $\mathbf{V}$, we can obtain two square stochastic matrices $\mathbf{W}_m = \mathbf{U} \times \mathbf{V}$ and $\mathbf{W}_n = \mathbf{V} \times \mathbf{U}$.
3. We calculate the eigenvalues λ_α and the corresponding eigenvectors $\vec{p}_\alpha$ of both $\mathbf{W}_m$ and $\mathbf{W}_n$;
4. We merge nodes with similar components in the $\vec{p}_\alpha$ as one node. In the new adjacency matrix $\tilde{A}$, this node will carry the sum of the edges of original nodes. The nodes in the top set should be merged based on the eigenvectors of $\mathbf{W}_m$ and the nodes in the bottom set should be merged based on the eigenvectors of $\mathbf{W}_n$.

Unlike the original network, the reduced network is a weighted one. Though the low-strength links play a less significant role in the random walk process, they must exit to make the reduced network connected. The new stochastic matrices $\tilde{\mathbf{U}}$ and $\tilde{\mathbf{V}}$ are calculated as $\tilde{U}_{ij} = \tilde{A}_{ij}/\sum_j \tilde{A}_{ij}$ and $\tilde{V}_{ij} = \tilde{A}_{ji}/\sum_j \tilde{A}_{ji}$. This method can be further extended to more than one eigenvector. In this case, groups are defined as nodes with almost the same components in the eigenvectors corresponding to the largest nontrivial eigenvalues. It turns out that choosing several largest nontrivial eigenvalues could better preserve the properties of random walk in bipartite network.

III. RESULTS

To validate our method, we apply it to both artificial and real-world bipartite networks.

A. Artificial networks

To begin with, we consider an artificial bipartite network with 1200 vertices which are divided into 2 sets. The top set has 300 vertices and the bottom set has 900 vertices. We divide nodes into 10 groups with the same size, such that each group has 30 vertices from the top set and 90 vertices from the bottom set. The probability for having a link between each

pair of nodes in the same group is q_1, and q_2 is the corresponding probability between groups. In this section, $q_1 = 0.4$ and $q_2 = 0.05$. Since this kind of artificial network has obvious community structure, we call them community networks. Using the clustering method in original bipartite network,[13] we can correctly detect 10 communities from the network.

To coarse grain this bipartite network, we used the eigenvectors ($\vec{p}_2$, $\vec{p}_3$, and $\vec{p}_4$) of the three largest nontrivial eigenvalues. We set $I = 12$, which means the interval between the largest and the smallest components of each eigenvector into 12 equal parts. Using the BSCG method, we get a rather small network with 391 vertices. Since the BSCG method and community detection methods focus on different properties of the bipartite network, the grouping results are different. Here, we compare the BSCG method to a typical CD method in Ref. 13. The random coarse graining (RCG) method is also carried out for comparison. Table I shows the three largest nontrivial eigenvalues of $\mathbf{W}_m$ before and after coarse graining (the three largest nontrivial eigenvalues of $\mathbf{W}_m$ and $\mathbf{W}_n$ are the same). Obviously, the largest three eigenvalues are effectively preserved by the BSCG method in the coarse-grained network. However, these eigenvalues are largely changed if the network is coarse grained by the CD or RCG method.

Moreover, we also apply the BSCG method to ER bipartite networks and obtain similar results (see also Table I). In ER bipartite networks, the probability for having a link between two vertices of different sets is 0.01 and the top set contains 1000 vertices while bottom set has 800 vertices. We also focus on the eigenvectors of the three largest nontrivial eigenvalues and set $I = 20$. The results in Table I indicate that our new method is robust in various kinds of artificial networks.

B. Real-world bipartite networks

In this subsection, we apply our method to some real-world networks. First, we use a social network of terrorists. The data, collected from 430 websites, were based on the relationship between terrorists and their organizations. The network was sampled from the data collected over a period from Oct. 1st, 1949 to May 1st, 2012. In this small social network, we focus on the giant component which is composed of 73 nodes in total, including 20 organizations and 53 people. The structure of the original network can be seen in the top figure of Fig. 1, where the blue squares account for people and the red circles represent the organizations. The links between two nodes indicate that a person belongs to an organization. To coarse grain this network, we set $I = 5$ and obtain a reduced network with 23 nodes, which is shown in the bottom figure of Fig. 1. The three largest nontrivial eigenvalues before and after coarse graining are reported in Table II. Clearly, all these eigenvalues are kept almost unchanged. Moreover, the bipartite structure of the original network is well preserved as shown in Fig. 1. We also try the method introduced in Ref. 21 on this real-world network, the resulting network is a monopartite one and the original two different kinds of nodes are indistinguishable.

TABLE I. The three largest nontrivial eigenvalues of $\mathbf{W}_m$ in the artificial networks including the bipartite network with community structure and the ER bipartite network. λ_α and $\tilde{\lambda}_\alpha$ are the eigenvalues before and after coarse graining, respectively.

Network	α	λ_α	$\tilde{\lambda}_\alpha$ (BSCG)	$\tilde{\lambda}_\alpha$ (CD)	$\tilde{\lambda}_\alpha$ (RCG)
Community network	2	0.4405	0.4336	0.4051	0.0924
	3	0.4342	0.4279	0.3920	0.0824
	4	0.4180	0.4076	0.3809	0.0792
ER network	2	0.3986	0.3933	0.1812	0.1097
	3	0.3908	0.3833	0.1717	0.1065
	4	0.3865	0.3784	0.1690	0.1037

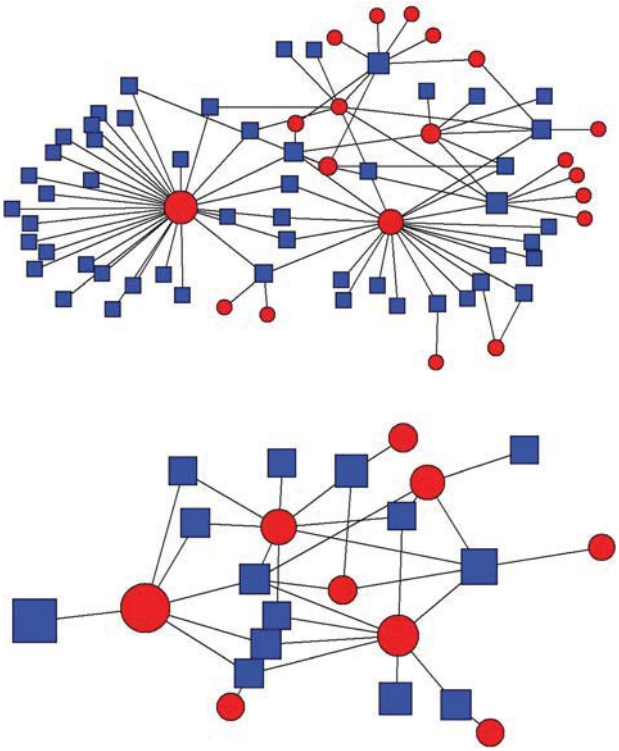

FIG. 1. The top figure is a social bipartite network of terrorists with $N + M = 73$. Nodes' size is proportional to their degree. The two different colors represent the two kinds of vertices. The blue squares stand for people and the red circles represent the organizations. The bottom figure is the coarse-grained network from the BSCG method with $N + M = 23$. Nodes' size is proportional to its strength in this weighted network.

As a further step, we apply our method to two online commercial networks: MovieLens and Netflix. The movielens network was sampled from the data collected over a seven-month period from September 19th, 1997 through April 22nd, 1998. The data consisted of 100 000 movie ratings from 943 users on 1682 items. Each user sampled had rated at least 20 items. Users can vote for movies with five level ratings from 1 (i.e., worst) to 5 (i.e., best). Here we only consider the ratings higher than 2, so that the final data contain 82 520 user-object pairs. This sampled data are freely available at Ref. 41. The Netflix network was randomly sampled from the very large data set provided for the Netflix Prize. The original data are freely available at Ref. 42. It has 480 189 users, 17 770 items and 100 480 507 ratings. In the paper, we only consider a subset of this very large data set. The subset consists of 3000 users, 2779 movies, and 824 802

TABLE II. The three largest nontrivial eigenvalues of $\mathbf{W}_m$ in real-world bipartite networks including a small terrorists' social network, movielens network, and Netflix network. λ_α and $\tilde{\lambda}_\alpha$ are, respectively, the eigenvalues before and after coarse graining.

Network	α	λ_α	$\tilde{\lambda}_\alpha$ (BSCG)	$\tilde{\lambda}_\alpha$ (CD)	$\tilde{\lambda}_\alpha$ (RCG)
Terrorists	2	0.8070	0.8059	0.7132	0.3781
	3	0.7259	0.7256	0.5639	0.2647
	4	0.6013	0.5732	0.5000	0.1868
Movielens	2	0.4180	0.4093	0.3195	0.0246
	3	0.2436	0.2305	0.1055	0.0173
	4	0.2075	0.1890	0.0864	0.0153
Netflix	2	0.2575	0.2535	0.1369	0.0139
	3	0.2209	0.2168	0.1313	0.0132
	4	0.2148	0.1971	0.1271	0.0115

links. Similarly to the MovieLens data, only the links with ratings no less than 3 are considered. After data filtering, there are 197 428 links left in the Netflix network.

We first investigate how these three nontrivial eigenvalues evolve when the nodes in the networks are merged. Fig. 2 shows the change of these eigenvalues as a function of network size $N+M$. The red line corresponds to a random merging of the nodes into groups, the green line is the result of community detection method, and the blue line shows the results of the BSCG method in which $\vec{p}_2$, $\vec{p}_3$, and $\vec{p}_4$ are considered. The different values of network size $N+M$ correspond to different choices of the number of intervals I. Generally speaking, a small I yields a small network size. As shown in Fig. 2, these three eigenvalues are well preserved in BSCG method even though the network size is significantly reduced. Actually, I can be regarded as a parameter to determine how accurate the eigenvalues are expected to be preserved, larger I can improve the precision of the method while resulting a bigger size of the reduced network.

In Fig. 2, it is also clearly shown that if nodes are merged randomly or according to the community detection method, the eigenvalues change dramatically. Consequently, the properties of random walk will be significantly changed. In order to keep eigenvalues almost unchanged, we set $I = 12$ in the BSCG method and get a reduced movielens network with size $N+M = 500$, which is 20% as big as the original network. In Netflix network, we set $I = 60$ and finally 657 nodes are left, which is about 10% as big as the size of the original network. The three largest eigenvalues in the both reduced networks can be seen in Table II.

A more direct test of our method is to compare the first passage time (MFPT) from node i to node j, which is denoted by T_{ij} in the original and reduced networks. We label the nodes in the bipartite network from 1 to $N'(N' = N+M)$ and consider the bipartite network as a monopartite one. In this way, all the cases for random walk in bipartite networks are included, i.e., the random walker can start from one type of nodes and finally arrive at either the same type or the other type of nodes. We consider the multi-sink random walk problem[34,39] and the MFPT can be exactly calculated by Eq. (3).

In order to compare the MFPT between the original and reduced networks in movielens, we use the coarse grained network with $N' = 500$ obtained above. Specifically, we consider that the walker starts at each node in the top set and define the node i with the largest strength as the sink in the

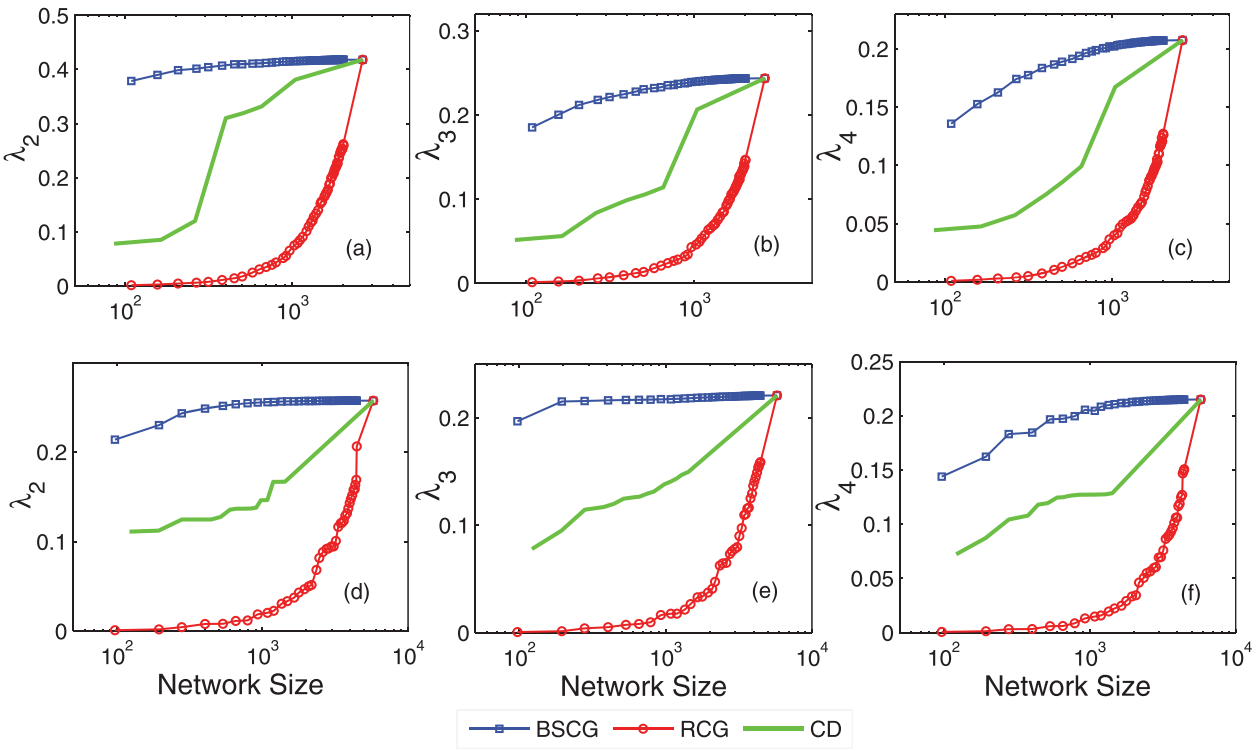

FIG. 2. The evolution of the three largest nontrivial eigenvalues λ_2, λ_3, and λ_4 as a function of the size of the coarse-grained network. (a)–(c) The original network is movielens network. (d)–(f) The original network is Netflix network. Red circles correspond to the random coarse graining method, the green line is the community detection method, and the blue squares represent the BSCG method.

bottom set. In Fig. 3(a), blue circles represent the MFPT from each node in the top set to nodes belonging to the group i in the bottom set in the original network. The MFPT to the group i in the bottom set in the reduced network is displayed with red lines. The exact overlap indicates that the MFPT is well preserved in the reduced network. The inset of Fig. 3(a) shows the relationship between the MFPT of the original network and that of the reduced network. The result implies almost equal MFPT in the original and the reduced network, given the same the source node and the sink. The slope of the curve is 0.996 and the goodness of linear fit is $R^2 = 0.998$. However, the random coarse graining method significantly destroys the MFPT. As shown in Fig. 3(b), the MFPT between original network and reduced network differs from each other. From the inset of Fig. 3(b), it is shown that there is no significant relationship between these two MFPT. Compared to the random coarse graining, the community detection performs slightly better. However, we can still observe that the red line and blue line do not overlap well. We further test the MFPT in the Netflix network and its coarse gained networks from BSCG method, CD, and RCG method. Similar results are obtained (see Figs. 3(d)–3(f)).

Besides computing the exact MFPT from Eq. (3), we also use the numerical simulation of the random walk process to test the BSCG method. Specifically, we put a walker on each node in the bipartite network and let it travel based on the stochastic matrices (**U** and **V**). Similar results to Fig. 3 are obtained, namely the reduced network from the BSCG method effectively preserves the MFPT while the CD and RCG methods significantly change the MFPT. Finally, we remark that the results in Fig. 3 are consistent in different choices of sinks. No matter whether the walker starts and ends at nodes in the same or different set of nodes, the MFPT line of the reduced network from BSCG method well overlaps with that of the original network.

In real application, the computational complexity of the method is a crucial factor. Any coarse graining method will become meaningless if the consuming time is unacceptable. Generally, the time complexity for calculating all the eigenvalues and eigenvectors of a matrix is $O(N^3)$. However, in our algorithm, we only use the largest three nontrivial eigenvectors. These eigenvectors are quite fast to calculate using the power method for sparse matrices,[6] in time $O(N^2)$. Even if we want to know all the eigenvalues and eigenvectors, combining the Lanczos and QL algorithms, these eigenvectors of a sparse symmetric matrix can be obtained in time $O(NE)$, where E is the number of links in the network. A similar method, the Arnoldi algorithm, can be used for an asymmetric matrix.[6] On the other hand, if we directly run the random walk process on the original network, the computational complexity for calculating the MFPT is $O(N^3)$. Therefore, our coarse graining method is meaningful in practical use, especially for large and sparse bipartite networks.

We finally consider the robustness of our method. Specifically, a robust spectral-based coarse graining method should be able to preserve the network function even when the considered eigenvalues cannot fully represent the properties of the whole network (i.e., the eigenvalues used for

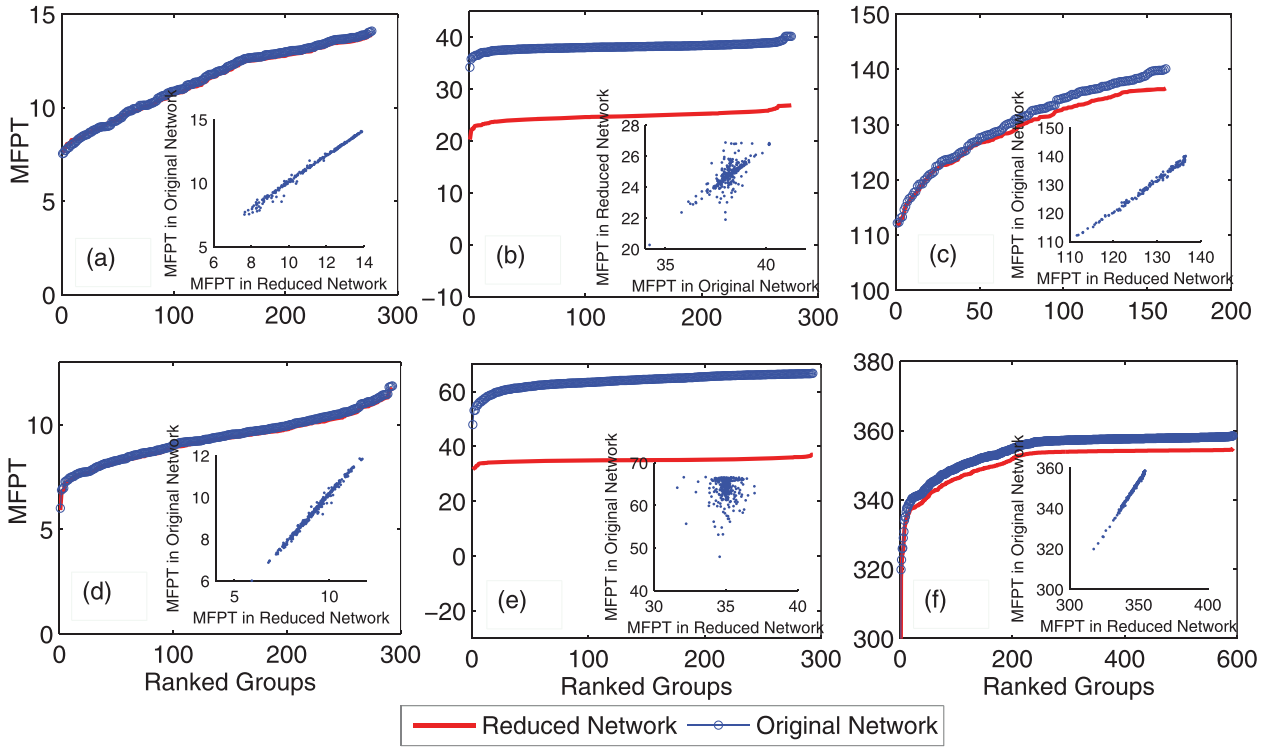

FIG. 3. Comparison of the MFPT. The walker starts at each node in the top set and the sink i is selected as the node with the strongest weight in the bottom set. The blue circles represent the average MFPT ranked for each group in the original network. The MFPT of the corresponding nodes in the coarse-grained network is displayed with red lines. (a) Nodes merged by BSCG method in Movielens network. (b) Nodes merged randomly in Movielens network. (c) Nodes merged based on community detection method in Movielens network. (d) Nodes merged by BSCG method in Netflix network. (e) Nodes merged randomly in Netflix network. (f) Nodes merged based on community detection method in Netflix network. Insets: Comparison of the exact MFPT between original and the reduced bipartite network. Slope 1 represents the well preserved MFPT in the reduced network.

coarse graining is not separated enough from the next few ones). We use the artificial network with community structure in Sec. III A to modify the gap between eigenvalues. The results show that the BSCG method can still effectively preserve the eigenvalues and MFPT when the size and location of the gap are changed.

IV. CONCLUSION

One of the most difficult hurdles in the analysis of complex network is the very large size of the real-world systems. If the network has more than 10^5 nodes, many algorithms are significantly slow and sometimes the application is even prohibitive. In order to solve this challenge, some coarse graining method for complex networks has been proposed. These methods mainly focus on the monopartite network in which only one type of nodes exist.

In this paper, we proposed a new coarse grain method for bipartite network. After introducing two square stochastic matrices $\mathbf{W}_m$ and $\mathbf{W}_n$, we find that their three largest nontrivial eigenvalues can effectively represent the properties of random walks. After merging nodes with similar components in the eigenvectors of these eigenvalues, the reduced network with well preserved eigenvalues of stochastic matrix is obtained. Moreover, a direct test based on the mean first passage time is carried out in two real-world bipartite networks, showing that this property is also well preserved in the reduced network. We believe that this method can be easily extended to preserve many other spectral-determined dynamical properties in bipartite networks. Finally, we remark that for a bipartite network the coarse graining provides a highly representative approximation of the initial network, resulting a way to circumvent the large size of complex networks for their analysis and visualization.

ACKNOWLEDGMENTS

This work was supported by the NSFC under Grant Nos. 61174150, 70771011, and 60974084, NCET-09-0228, and fundamental research funds for the Central Universities of Beijing Normal University. Y.W. thanks the China Scholarship Council (CSC) for support.

[1]R. Albert and A.-L. Barabási, Rev. Mod. Phys. **74**, 47 (2002).
[2]M. E. J. Newman, SIAM Rev. **45**, 167 (2003).
[3]A.-L. Barabási and R. Albert, Science **286**, 509 (1999).
[4]D. J. Watts and S. H. Strogatz, Nature **393**, 440 (1998).
[5]S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D.-U. Hwang, Phys. Rep. **424**, 175 (2006).
[6]M. E. J. Newman, *Networks: An Introduction* (Oxford University Press, 2010).
[7]A. Zeng, C. H. Yeung, M.-S. Shang, and Y.-C. Zhang, Europhys. Lett. **97**, 18005 (2012).
[8]C.-J. Zhang and A. Zeng, Physica A **391**, 1822 (2012).
[9]M. E. J. Newman, Phys. Rev. E **64**, 016131 (2001).
[10]M. E. J. Newman, Phys. Rev. E **64**, 016132 (2001).
[11]T. S. Evans and A. D. K. Plato, Phys. Rev. E **75**, 056101 (2007).
[12]P. G. Lind, M. C. Gonzalez, and H. J. Herrmann, Phys. Rev. E **72**, 056127 (2005).
[13]P. Zhang, J. Wang, X. Li, M. Li, Z. Di, and Y. Fan, Physica A **387**, 6869 (2008).
[14]M. Kitsak and D. Krioukov, Phys. Rev. E **84**, 026114 (2011).
[15]B. Bollobás, *Graph Theory and Combinatorics* (Academic, London, 1984), p. 35.
[16]M. Kitsak, L. K. Gallos, S. Havlin, F. Liljeros, L. Muchnik, H. E. Stanley, and H. A. Makse, Nat. Phys. **6**, 888 (2010).
[17]C. Song, S. Havlin, and H. A. Makse, Nature **433**, 392 (2005).
[18]K.-I. Goh, G. Salvi, B. Kahng, and D. Kim, Phys. Rev. Lett. **96**, 018701 (2006).
[19]B. J. Kim, Phys. Rev. Lett. **93**, 168701 (2004).
[20]D. Gfeller and P. De Los Rios, Rhys. Rev. Lett. **100**, 174104 (2008).
[21]D. Gfeller and P. De Los Rios, Phys. Rev. Lett. **99**, 038701 (2007).
[22]K. Rajendran and I. G. Kevrekidis, Phys. Rev. E **84**, 036708 (2011).
[23]H. Chen, Z. Hou, H. Xin, and Y. Yan, Phys. Rev. E **82**, 011107 (2010).
[24]M. Girvan and M. E. J. Newman, Proc. Natl. Acad. Sci. U.S.A. **99**, 7821 (2002).
[25]M. E. J. Newman, Phys. Rev. E **74**, 036104 (2006).
[26]J. Duch and A. Arenas, Phys. Rev. E **72**, 027104 (2005).
[27]M. E. J. Newman, Proc. Natl. Acad. Sci. U.S.A. **103**, 8577 (2006).
[28]Y. Hu, Y. Nie, H. Yang, J. Cheng, Y. Fan, and Z. Di, Phys. Rev. E **82**, 066106 (2010).
[29]B. Karrer, E. Levina, and M. E. J. Newman, Rhys. Rev. E **77**, 046119 (2008).
[30]A. Lancichinetti, F. Radicchi, and J. J. Ramasco, Phys. Rev. E **81**, 046110 (2010).
[31]S. Fortunato, Phys. Rep. **486**, 75 (2010).
[32]A. Zeng and L. Lü, Phys. Rev. E **83**, 056123 (2011).
[33]S. Lehmann, M. Schwartz, and L. K. Hansen, Phys. Rev. E **78**, 016108 (2008).
[34]J. D. Noh and H. Rieger, Phys. Rev. Lett. **92**, 118701 (2004).
[35]J. G. Kemeny and J. L. Snell, *Finite Markov Chains* (Springer, New York, 1976).
[36]D. Aldous and J. Fill, *Reversible Markov Chains and Random Walks on Graphs* (University of California, Berkeley, 2002). Available at: http://www.stat.berkeley.edu/~aldous/RWG/Chap2.pdf.
[37]T. Zhou, Z. Kuscsik, J.-G. Liu, M. Medo, J. R. Wakeling, and Y.-C. Zhang, Proc. Natl. Acad. Sci. U.S.A. **107**, 4511 (2010).
[38]L. Lü and W. Liu, Phys. Rev. E **83**, 066119 (2011).
[39]A. Baronchelli and V. Loreto, Phys. Rev. E **73**, 026103 (2006).
[40]Z. Zhang, Y. Yang, and Y. Lin, Phys. Rev. E **85**, 011106 (2012).
[41]See www.grouplens.org for information about the movielens data.
[42]See www.netflixprize.com for information about the netflix data.