Physica A 513 (2019) 424–437

Contents lists available at ScienceDirect

Physica A

journal homepage: www.elsevier.com/locate/physa

A spectral method to find communities in bipartite networks

Liang Feng [a,b], Cangqi Zhou [c], Qianchuan Zhao [a,b,*]

[a] Center for Intelligent and Networked Systems (CFINS), Department of Automation, Tsinghua University, Beijing 100084, China
[b] Tsinghua National Laboratory for Information Science and Technology (TNList), Tsinghua University, Beijing 100084, China
[c] School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China

HIGHLIGHTS

- Optimizing Barber's bipartite modularity is represented as a spectral problem.
- A new method is proposed to obtain the community structure of bipartite network.
- The new method helps to alleviate the resolution-limit issue.

ARTICLE INFO

Article history:
Received 17 April 2018
Received in revised form 6 August 2018
Available online 5 September 2018

Keywords:
Community detection
Bipartite networks
Singular value decomposition
Modularity optimization

ABSTRACT

Community detection in complex networks that aims to find partitions of networks with dense intra-edges and sparse inter-edges, has recently attracted lots of interest in many fields. Specially, bipartite networks composed of two different types of vertices are the common representations for many real-world networks, such as actor–film, consumer–product networks, etc. In this paper, we show that optimizing Barber's bipartite modularity, which is widely used to evaluate partitions of bipartite networks, can be reformulated as a spectral problem with appropriate relaxations. We further propose a new method combining singular value decomposition(SVD) and BRIM algorithm to obtain an optimal community partition. Compared with many other algorithms, the new method can give us a more detailed and comprehensive view of the original bipartite network for different cluster numbers k. We test our method on both synthetic networks and two benchmark data sets. Experimental results show that, our method is not only capable to extract a community partition with a larger bipartite modularity, but also converge to the exact underlying community partition when k is appropriately set, which helps to alleviate the resolution limit issue to some extent.

1. Introduction

Many real-world systems can be represented as networks, in which vertices are usually not uniformly distributed but naturally clustered into small groups. Community detection aims to find partitions of networks with dense intra-edges and sparse inter-edges [1]. Vertices in the same cluster may share some common properties and different clusters usually play different roles in real networks. Therefore, community detection is useful to similar modules identification, node classification and so on [2]. As a result, community detection has recently attracted lots of interest in many fields including social networks [1,3] and biological networks [2].

There are many algorithms designed to identify the community structure of complex networks, and most of them optimize a specific objective to get the optimal or near-optimal partitions. One well-known method is spectral clustering,

* Corresponding author at: Center for Intelligent and Networked Systems (CFINS), Department of Automation, Tsinghua University, Beijing 100084, China.

E-mail addresses: l-feng16@mails.tsinghua.edu.cn (L. Feng), zhaoqc@tsinghua.edu.cn (Q. Zhao).

https://doi.org/10.1016/j.physa.2018.09.022
0378-4371/

in which the objective is usually to optimize a global cost function, such as Ratio Cut and Normalized Cut. By discarding the discreteness condition of cost functions, the problem can be solved by the eigenvectors of the (un)normalized Laplacian matrix of graphs [4]. Though it only needs a pairwise similarity matrix of original network vertices, there are some drawbacks of spectral clustering. First, implicitly determined by the cost functions, spectral clustering usually tries to find balanced clusters, which is not consistent with the heterogeneous community sizes in many real complex networks [1,2]. Second, many spectral clustering algorithms need to know a prior partition number k, since optimized objectives which are biased by k increase (or decrease) monotonically as k varies [4,5]. Another type of methods usually relies on a famous criterion called *modularity* [6], which measures the difference between the actual edges within communities and the expectation of edges derived from a null model. The null model keeps the heterogeneous degrees of vertices, but assumes edges are randomly connected between vertices. Therefore, larger values of modularity usually indicate stronger community structures. Modularity has become a popular evaluation criterion of network partitions. Many optimization algorithms [7–11] as well as some modified spectral methods [12–14] have been developed in recent years. But it still has some limitations such as the resolution-limit issue, which makes it hard to find small clusters in some cases [15,16].

Bipartite network is a significant class of network. There are two different types of vertices in bipartite networks, and two vertices of same type are nonadjacent. Above features make bipartite network provide a natural representation for many real-world networks, such as actor–film, consumer–product networks, etc. A spectral method is proposed by Dhillon to co-cluster documents and words in bipartite networks. It uses singular value decomposition to solve the relaxed minimum Normalized Cut problem [17]. Similar to the development of community detection of unipartite network, Barber proposed a bipartite modularity. It stems from the original Newman's unipartite modularity and aims to measure the community partitions of bipartite networks. Besides, he proposed a recursive algorithm called BRIM to identify the community structure [18]. Since then, numerous methods have been developed to maximize the bipartite modularity, including label-propagation [19,20] method and bipartite Louvain algorithm [21].

In this paper, we develop a new spectral method to optimize Barber's bipartite modularity. We first derived the spectral form of optimizing bipartite modularity with appropriate relaxations given cluster num k, as White did in [12] for unipartite modularity. The solutions are exactly the eigenvectors of singular value decomposition of bipartite modularity matrix. However, the left and right eigenvectors are corresponding to *column space* and *row space* of original matrix, co-clustering all the vertices is not suitable. This situation is different from [17]. Since the differences among vectors of similar vertices are small, we initially get a partition of one part vertices (we choose the smaller part) using k-means. After that, we further use BRIM algorithm to assign community indexes of the other part and iteratively maximize bipartite modularity. We test the new method on both synthetic networks with (un)equal sized bipartite cliques and two benchmark data-sets. Experimental results show that, our method can obtain a community partition with a larger bipartite modularity than many other existing algorithms. When there is no local optimal partition for a specific prior partition number k, the method can converge to a nearby stable partition of the original network. The convergence makes the new method helpful to alleviate the resolution-limit problem, since bipartite modularity usually decreases monotonically as k increases when k is larger than the underlying cluster number.

The structure of the remainder of this paper is as follows. In Section 2, we will introduce the definition of modularity in both unipartite and bipartite networks, and show how to reformulate the problem of optimizing bipartite modularity as a spectral issue given a priori cluster num k. In Section 3, we will give a brief introduction of singular value decomposition and BRIM algorithm, and show the details of new method along with complexity analysis. In Section 4, our method is evaluated on synthetic networks and two benchmark data-sets. Finally, it concludes with a summary and discussion in Section 5.

2. Reformulation of optimizing bipartite modularity

In general, the community structure of network refers to a partition of all the vertices, in which each vertex belongs to one cluster (non-overlapping partition) or more (overlapping partition) clusters. In this paper, we only consider the non-overlapping partition. Usually, a good partition means that the density of intra-edges within communities are higher than the density of inter-edges between them [1]. Although there is no common definition of community [2], modularity is widely used in recent decades.

In this section, we first review the definition of modularity, both in unipartite and bipartite networks. Then given the cluster num k is already known, we show the strategies of how to reformulate the problem of optimizing Barber's bipartite modularity as a spectral issue.

2.1. Unipartite modularity

Mathematically, a network can be modeled as a graph composed of vertices V and edges E. For simplicity, we only discuss undirected and unweighted graphs without self-loops nor multiple edges in the following section, but our discussion can be easily extended to directed and weighted graphs. Considering a graph $\mathcal{G}$ with n vertices and m edges is defined by a $n \times n$ adjacent matrix $A := (A_{ij})$, where A_{ij} equals 1 if there is an edge connecting vertex i and j, and 0 otherwise. Each vertex i is assigned to a community denoted by c_i, and the degree of vertex i is the sum of edges incident to it which is denoted as $k_i = \sum_j A_{ij}$. The unipartite modularity [6] is defined as

$$Q = \frac{1}{2m} \sum_{i,j} \Big(A_{ij} - \frac{k_i k_j}{2m} \Big) \delta(c_i, c_j)$$

where A_{ij} is the element of adjacent matrix A, $\frac{k_i k_j}{2m}$ is the expected edge between vertex i and j in the null model which keeps the heterogeneous degree of each vertex, function $\delta(c_i, c_j)$ is 1 if $c_i = c_j$, and 0 otherwise. It is easy to see that, larger value of modularity indicates there are more edges than expected within communities, which naturally implies a stronger community structure. The original formulation of modularity considers all pairs of vertices in the network, alternatively we can compute modularity in another way. Assume $\mathcal{P}_k$ is a partition in which all the vertices are divided into k clusters. For two vertices i and j, term $\delta(c_i, c)\delta(c_j, c)$ is 1 if they both belong to community c, and 0 otherwise. For a certain community c, intra-edges is $\sum_{i,j} A_{ij}\delta(c_i, c)\delta(c_j, c)$. Since both ijth and jith element of A contribute to community intra-edges, $\sum_{i,j} A_{ij}\delta(c_i, c)\delta(c_j, c)$ is twice of the sum of edges in community c. Similarly, $\sum_i k_i\delta(c_i, c)$ is the sum of degrees of all the vertices in community c, and $\sum_{i,j} k_i k_j \delta(c_i, c)\delta(c_j, c)$ is the product of $\sum_i k_i\delta(c_i, c)$ and $\sum_j k_j\delta(c_j, c)$. Then the modularity can be rewritten as

$$Q(\mathcal{P}_k) = \sum_{c=1}^{k}\left[\frac{A(V_c, V_c)}{2m} - \left(\frac{A(V_c, V)}{2m}\right)^2\right] \tag{1}$$

where V is the set of vertices in $\mathcal{G}$, V_c is the set of vertices in c, $A(V_c, V_c)$ is the double of intra-edges of community and $A(V_c, V)$ is the sum of degrees of all the vertices in community c.

2.2. Bipartite modularity

Bipartite network is a special type of unipartite networks whose vertices can be divided into two disjoint subsets. There are no edges existing between two vertices in the same subset. Equivalently, we can use two different colors, say red and blue, to assign every vertex in a bipartite network, with the constrain that no neighboring vertices share the same color. Assume X is the red subset of a bipartite network $\mathcal{H}$, p is the number of red vertices, Y is the blue subset, q is the number of blue vertices and m is the number of total edges. The total vertices in the bipartite network is $n = p + q$ and without loss of generality, we can assume the indexes of red vertices are $1, 2, \ldots, p$ and blue vertices are labeled $p + 1, p + 2, \ldots, p + q$. The $(p + q) \times (p + q)$ adjacent matrix of $\mathcal{H}$ has a block off-diagonal form

$$A_{(p+q)\times(p+q)} = \begin{bmatrix} O_{p\times p} & B_{p\times q} \\ B^T_{q\times p} & O_{q\times q} \end{bmatrix}$$

where $O_{p\times p}$ is a $p \times p$ null matrix. $B_{p\times q}$ is the bipartite adjacent matrix representing the edges between red vertices and blue vertices, where B_{ij} is 1 if red vertex i is adjacent to blue vertex j, and 0 otherwise. It is obvious that matrix B contains all information of edges in $\mathcal{H}$.

Similar to Newman's unipartite modularity, Barber [18] proposed a bipartite modularity to evaluate partitions of bipartite network. Barber's bipartite modularity avoids the drawbacks of common used definitions of community structure of bipartite network and one-mode projection method [22,23]. Since edges of bipartite network only exist between two vertices with different colors, Barber defined a bipartite null model. In the null model, the expected number of edges between red vertex i and blue vertex j is $\frac{d_i g_j}{m}$, where $d_i = \sum_j^q B_{ij}$ is the degree of red vertex i and $g_j = \sum_i^p B_{ij}$ is the degree of blue vertex j. The bipartite modularity is defined as

$$Q_b = \frac{1}{m}\sum_{i=1}^{p}\sum_{j=1}^{q}\left(B_{ij} - \frac{d_i g_j}{m}\right)\delta(c_i, c_j)$$

where c_i, c_j indicate the communities to which vertices i, j belong. Term $\delta(c_i, c)$ is 1 if vertex i belongs to community c, and 0 otherwise. If we reformulate bipartite modularity from the perspective of communities, the edges inside community c are $\sum_{i,j} B_{ij}\delta(c_i, c)\delta(c_j, c)$ which represent the edges between red vertices and blue vertices. Term $\sum_i^p d_i x_{ic}$ indicates the total degrees of red vertices inside c, and $\sum_{i,j} d_i g_j \delta(c_i, c)\delta(c_j, c)$ is actually the product of degrees of all red vertices and degrees of all blue vertices in c. The bipartite modularity can be rewritten as

$$Q_b(\mathcal{P}_k) = \frac{1}{m}\sum_{c=1}^{k}\left[B(V_c^{\mathrm{r}}, V_c^{\mathrm{b}}) - \frac{B(V_c^{\mathrm{r}}, V^{\mathrm{b}})B(V_c^{\mathrm{b}}, V^{\mathrm{r}})}{m}\right] \tag{2}$$

where V is the set of all vertices in $\mathcal{H}$, V_c^{r} and V_c^{b} are the red and blue vertices set in community c, $B(V_c^{\mathrm{r}}, V_c^{\mathrm{b}})$ is the sum of edges within community, $B(V_c^{\mathrm{r}}, V^{\mathrm{b}})$ and $B(V_c^{\mathrm{b}}, V^{\mathrm{r}})$ are the sum of degrees of all red and blue vertices in community c, respectively.

2.3. Spectral formulation of optimizing bipartite modularity

In this part, we develop the spectral formulation of optimizing bipartite modularity given the cluster num k is already known. White et al. [12] has already shown that the Newman's modularity measure can be related to a spectral clustering method. Structurally, the discussion below parallels with the deduction of spectral formulation of unipartite modularity.

For simplicity, we only discuss unweighted bipartite graphs without self-loops nor multiple edges in the following section. We follow two steps to reformulate the problem of optimizing Barber's bipartite modularity. First, reconstruct the original problem as a discrete quadratic assignment problem. Second, relax the condition of discreteness to continuous assignment and use lagrangian multiplier method to solve the resulting problem. In the following part, we will describe both steps in detail.

Discrete Quadratic Assignment. Considering a connected bipartite network $\mathcal{H}$ with a path between any two vertices, there is a k-partition $\mathcal{P}_k$ of vertices, where every vertex i belongs to one and only one community. The community indexes of vertices are taken from $1, 2, \ldots, k$. Alternatively, the community indexes of all red vertices can be represented as a $p \times k$ matrix $X = [x_1, \ldots, x_k]$, where x_{ic} is 1 if red vertex i belongs to community c, and 0 otherwise. We can also define a $q \times k$ community index matrix $Y = [y_1, \ldots, y_k]$ for blue vertices. In order to describe the notations clearly, we first rewrite Eq. (2) as follows

$$Q_b(\mathcal{P}_k) \propto \sum_{c=1}^{k}\left[mB(V_c^{\mathrm{r}}, V_c^{\mathrm{b}}) - B(V_c^{\mathrm{r}}, V^{\mathrm{b}})B(V_c^{\mathrm{b}}, V^{\mathrm{r}})\right] \tag{3}$$

where we multiply the original modularity formula by m^2.

Let $\boldsymbol{d} \in \mathcal{R}^{p\times 1}$ be a column vector representing the degrees of all red vertices where $d_i = \sum_j B_{ij}$, column vector $\boldsymbol{g} \in \mathcal{R}^{q\times 1}$ representing the degrees of all blue vertices where $g_j = \sum_i B_{ij}$. Since each vertex can only be in one community, it is obvious that $X\mathbf{1}_k = \mathbf{1}_p$ and $Y\mathbf{1}_k = \mathbf{1}_q$ where $\mathbf{1}_k$ is a all-ones column vector. If the actual and expected intra-edges are expressed in the form of community indexes, we can reformulate Eq. (3) as

$$\begin{aligned} Q_b(\mathcal{P}_k) &\propto \sum_{c=1}^{k}\left[m\sum_{i}^{p}\sum_{j}^{q} B_{ij}x_{ic}y_{jc} - \sum_{i}^{p} d_i x_{ic}\sum_{j}^{q} g_j y_{jc}\right] \\ &= \sum_{c=1}^{k}\left[m x_c^T B y_c - (d_i^T x_c)(g_j^T y_c)\right] \end{aligned}$$

where $\sum_i^p \sum_j^q B_{ij}x_{ic}y_{jc}$ is the number of actual edges and $\sum_i^p d_i x_{ic} \sum_j^q g_j y_{jc}$ is the number of expected edges multiplied by m in community c. If we sum over all communities and use the facts that $\sum_{c=1}^k x_c^T B y_c = \mathrm{tr}(X^T BY)$ and $\sum_{c=1}^k (d_i^T x_c)(g_j^T y_c) = \mathrm{tr}(X^T d_i g_j^T Y)$ for community index matrixes X and Y, bipartite modularity can be further reduced as

$$\begin{aligned} Q_b(\mathcal{P}_k) &\propto \mathrm{tr}(X^T mBY) - \mathrm{tr}(X^T d_i g_j^T Y) \\ &= \mathrm{tr}\big(X^T(mB - d_i g_j^T)Y\big) \\ &= \mathrm{tr}\big(X^T(\mathcal{B} - \mathcal{D})Y\big) \end{aligned}$$

where $\mathcal{B} = mB$ and $\mathcal{D} = d_i g_j^T$. Combined with the constrain of hard-partition of vertices, maximizing bipartite modularity can be reformulated as

$$\begin{aligned} &\max_{X,Y}\{\mathrm{tr}\big(X^T(\mathcal{B} - \mathcal{D})Y\big)\} \\ &\text{s.t. } X^T X = M_r,\ Y^T Y = M_b \end{aligned} \tag{4}$$

where M_r and M_b are both $k \times k$ diagonal matrixes with diagonal elements $[M_r]_{cc}$ and $[M_b]_{cc}$ represent the number of red and blue vertices in community c.

Spectral Approximation. Since every element of community index matrix is either 0 or 1, this makes maximizing Eq. (4) be a NP-hard problem. Though it is difficult to find the optimal solution of original community index matrixes X and Y, we can use an ingenious way by relaxing the discreteness constrain by substituting $x_{ic}, y_{jc} \in \mathcal{R}^1$ for $x_{ic}, y_{jc} \in \{0, 1\}$. This is commonly used in spectral methods [4,12,13,17]. For the relaxed continuous optimization problem, we can rearrange Eq. (4) by means of lagrangian multiplier method as

$$\mathrm{tr}\big(X^T(\mathcal{B} - \mathcal{D})Y\big) + \Lambda(X^T X - M_r) + \Delta(Y^T Y - M_b) \tag{5}$$

where $\Lambda, \Delta \in \mathcal{R}^{k\times k}$ are diagonal matrixes. To get the optimal relaxed X and Y, take the derivatives of Eq. (5) with respect to X and Y and set them to 0. By rearranging relative terms, the conditions of optimal solution become

$$\begin{aligned} (\mathcal{B}_Q)Y &= 2X\Lambda \\ (\mathcal{B}_Q)^T X &= 2Y\Delta \end{aligned} \tag{6}$$

where $\mathcal{B}_Q = \mathcal{D} - \mathcal{B}$. So far, we have got the conditions which optimal X and Y satisfy. It is easy to prove that this problem can be solved by singular value decomposition, which we will discuss in the next section.

3. New method to cluster bipartite network

In Section 2, we have shown the strategies to reformulate optimizing bipartite modularity as a spectral issue and derived equilibrium equations to get the optimal relaxed X and Y. In this section, we will first explain how to use singular value decomposition to solve Eq. (5) and then introduce BRIM algorithm to further maximize the bipartite modularity.

3.1. Singular value decomposition (SVD)

SVD is a matrix factorization method commonly used to get the singular-value decomposition of a real matrix [24]. Given a $m \times n$ matrix M with $m \geq n$, the formal factorization form of SVD is defined as

$$M = U\Sigma V^T$$

where U is a $m \times m$ matrix, Σ is a $m \times n$ rectangular diagonal matrix with non-negative entries $(\sigma_1, \sigma_2, \ldots, \sigma_n)$ on the diagonal, and V is a matrix of dimensions $n \times n$. The diagonal entries of Σ are the singular values of M, and the columns of U and V are known as the left-singular and right-singular vectors of M. One important property of SVD is

$$U^T U = I_m, VV^T = I_n$$

where I_m and I_n are identity matrixes with appropriate dimensions, which means U and V are column orthogonal matrixes.

The diagonal entries of Σ also have the property that $\sigma_1 \geq \sigma_2 \geq \cdots \sigma_n \geq 0$. Let r be the rank of matrix M, then $\sigma_{r+1} = \sigma_{r+2} = \cdots = \sigma_n = 0$, which means the number of non-zero entries of Σ equals the rank r. Hence, the effective dimensions of these three matrixes are $m \times r$, $r \times r$ and $n \times r$, respectively. The r columns of U are corresponding to the orthogonal eigenvectors of MM^T which span the *column space*. The r columns of V are associated with the orthogonal eigenvectors of $M^T M$ which span the *row space* of matrix M. If we extract the first k columns of U and V to construct U_k and V_k, and the largest k singular values of Σ to construct Σ_k with all zeros entries on the off-diagonal. We can get the best *low rank* linear approximation of original matrix M

$$M = U\Sigma V^T \approx U_k \Sigma_k V_k^T$$

where $\min(m, n) \geq r \geq k$ and $U_k^T U = V_k^T V = I_k$. It is easy to reformulate above equation as

$$\begin{aligned} MV_k &\approx U_k \Sigma_k \\ M^T U_k &\approx V_k \Sigma_k \end{aligned} \tag{7}$$

Comparing Eq. (6) with Eq. (7), we can get the best linear approximations of the optimal relaxed X and Y:

$$X \approx U_k, Y \approx V_k, \Lambda = \Delta \approx \frac{1}{2}\Sigma_k \tag{8}$$

Therefore, the community index matrixes X and Y can be approximated by the first k eigenvectors of *column space* and *row space* of matrix $\mathcal{B}_Q$.

3.2. BRIM algorithm

Along with the definition of bipartite modularity, Barber [18] proposed an iterative algorithm called BRIM(bipartite, recursively induced modules). For a k partition of the original network, the algorithm first assumes all blue (or red) vertices are divided into modules through some method. Then community index matrix Y is fixed, and bipartite modularity can be represented as

$$Q_b(\mathcal{P}_k) = \frac{1}{m}\sum_{i=1}^{p}(\sum_{c=1}^{k} X_{ic}\tilde{Y}_{ic})$$

where $\tilde{Y} = \mathcal{B}_Q Y$. Since there is only a single 1 in each row of X and other elements are 0, we can just assign red vertex i to community c, where $\tilde{Y}_{ic}$ is the maximum of the ith row of $\tilde{Y}$. Similarly, when all red vertices are already assigned to modules, the modularity can be expressed as

$$Q_b(\mathcal{P}_k) = \frac{1}{m}\sum_{j=1}^{q}(\sum_{c=1}^{k} Y_{jc}\tilde{X}_{jc})$$

where $\tilde{X} = \mathcal{B}_Q^T X$. For this case, we can maximize bipartite modularity by assigning blue vertex j to community c where $\tilde{X}_{jc}$ is the maximum of the jth row of $\tilde{X}$. Taking the two steps together, we can induce a network partition from one initial division of blue vertices. The process iteratively increases bipartite modularity by inducing division of vertices, and stops at a local maximum. It should be noted that the final partition of network may have less than k communities, since there may be some communities containing vertices of only one type and the community will vanish during the process of iteration.

The performance of BRIM algorithm mainly depends on the initial partition of network. Random initialization of vertices is simple, but the performance is often relatively unstable and poor. Good initial partitions can motivate BRIM to get a better network partition, while a bad initial partition will make the algorithm get stuck in a poor local maximum quickly. Notice that BRIM algorithm also needs to know a priori partition number k to start the iteration process. In order to find the best k, Barber further put forward *adaptive* BRIM algorithm which adopts the random initialization and *bi-section* strategy to search a better k. Still, adaptive BRIM needs to try different values of k to get a relatively good partition. Afterwards, a method called LP&BRIM combining label-propagation and BRIM algorithm is proposed in [20]. The method uses the result of label-propagation as the initial partition and iteratively optimizes partition by BRIM algorithm.

There are mainly two drawbacks of these algorithms. First, since optimizing bipartite modularity is a NP-hard problem, there may exist many local maximums for different or even the same k. It will be time-consuming to use adaptive BRIM to get a good partition by trying different k for many times, since it uses random assignment initially. The result of label-propagation often centers on specific range of k, which also makes LP&BRIM hard to find global maximum. Second, sometimes we may want a specific k or near k partition of the original network. The random assignment often makes the result of adaptive BRIM relatively unstable, and the property of label propagation makes LP&BRIM unsuitable. Though bipartite Louvain [21] can get a hierarchical structure of original network, the partition number is discrete and resolution-limit issue often makes it get relatively large communities at the first step [15].

3.3. The new method

As is clear from the above descriptions, the optimal relaxed community index matrixes X and Y are orthogonal bases which span the *column space* and *row space* of $\mathcal{B}_Q$. They correspond to eigenvectors of matrix $\mathcal{B}_Q\mathcal{B}_Q^T$ and $\mathcal{B}_Q^T\mathcal{B}_Q$, respectively. If we extract the first k columns of U, then U_{ic} can be considered as the possibility of red vertex i in community c. The same meaning applies to the element of V_k. However, U and V are actually singular vectors of different matrixes, it is unsuitable to co-cluster red and blue vertices by just concatenating them.

In this article, we propose a new method to cluster bipartite network which combines singular value decomposition(SVD) and BRIM algorithm. As the following experimental results show that, new method can not only find a good community partition with larger bipartite modularity, but also can give us a more detailed and comprehensive view of the bipartite network from different cluster numbers k. Suppose we are seeking up to a maximum of K clusters where $K \leq r$, detailed steps of the new method are shown in Algo. 1 and algorithm flowchart is provided in Fig. B.8.

Algorithm 1: Our Proposed New Method.

Input: an initial bipartite network
Output: k with the largest bipartite modularity and corresponding partition

1. Compute matrix $\mathcal{B}_Q$, the number of red vertices p and blue vertices q.
2. Compute vector matrix $Q_K = U_K$ if $p \leq q$, and $Q_K = V_K$ otherwise.
3. For each value of k, $2 \leq k \leq K$:
   a. Form the matrix Q_k by extracting the first k columns of Q_K.
   b. Scale each row of Q_k by l^2-norm so that they all have unit length.
   c. Cluster the scaled row vectors of Q_k using k-means to get initial partition.
   d. Use BRIM algorithm to assign the other vertices and further iteratively optimize Q_b.

In step (2), we choose to only partition vertices of the smaller one, which means we just cluster red vertices if $p \leq q$, and blue vertices otherwise. There are two reasons for doing this. On one hand, a community is meaningless whose bipartite modularity is zero if all members are of one color, and SVD also shows that the rank of $\mathcal{B}_Q$ is no larger than the smaller of p and q. Therefore, the max number of communities is at most equal to p if $p \leq q$ and vice versa. On the other hand, there are many unbalanced bipartite networks in real-world such as the consumer–product network in which the number of categories is much smaller than consumers [21]. Besides that, if p is smaller, we can just compute the left-singular vectors of $\mathcal{B}_Q$ which are also the eigenvectors of $\mathcal{B}_Q\mathcal{B}_Q^T$ to save the running time. In step (3), we use the traditional k-means which is a very fast algorithm commonly used in vector-based clustering. The relationship between the clusters found by SVD and those obtained by first projecting the system onto unipartite networks is supplied in Appendix A. After that, we use the division result obtained in step (c) as initial partition and continue to optimize bipartite modularity by running BRIM algorithm.

3.4. Computational complexity

Computing a full SVD of a $p \times q$ matrix is fundamentally a $O(pq \cdot \min(p, q))$ problem, which may not be feasible to get the decompositions of extremely large matrixes. However, there are different algorithms with different complexities developed for SVD in the literature. For example, methods that calculate the rank-r partial SVD have reduced the complexity to $O(pqr)$ where $r \leq \sqrt{\min(p, q)}$ [25,26]. Also it should be noted that the decomposition is performed for matrix $\mathcal{B}_Q$ only once in step (2). In order to find optimal or suboptimal partitions for different values k, we only need to repeat step (3) where standard k-means and BRIM algorithm are performed.

Standard k-means with a Euclidean distance metric has complexity $O(ndke)$, where n is the number of data points, d is the dimensionality of each point which in the new method is equal to k and e is the number of iterations. But one can find much faster versions in the literature. In this article, we adopt the method which has a complexity of $O(nke)$ [27]. It uses geometric equalities to significantly reduce the computation of distance. There is also a modified version with the complexity of $O(nk)$ for large data, which incorporates approximate nearest neighbors search [28]. Therefore in the new method, when k iterates from 2 to K, the resulting complexity is roughly $O(nK^2e)$.

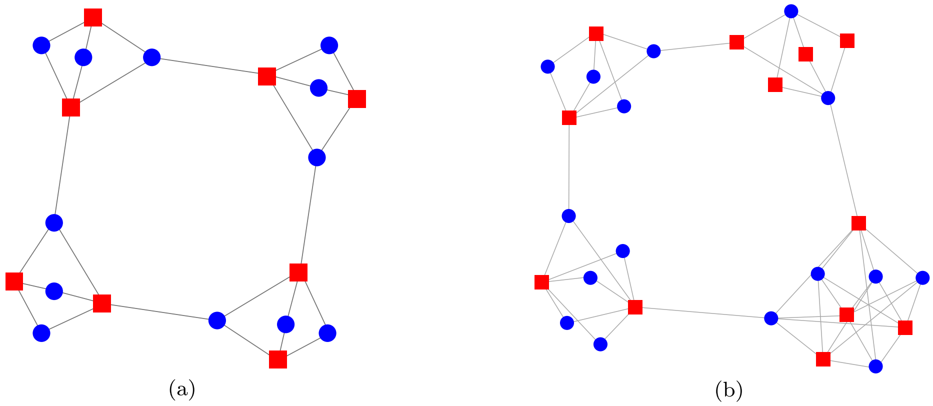

Fig. 1. Rings of bipartite cliques. (a) A bipartite network consists of 4 bipartite cliques with inter-edges from head to tail. (b) A bipartite network consists of 4 bipartite cliques of different sizes with the same rules.

For BRIM algorithm, the initial partition contains k communities. Assume $p \le q$, when inducing the division of blue vertices, we assign each blue vertex j to community c where $\tilde{X}_{jc}$ is the maximum of the jth row of $\tilde{X}$. This requires a time complexity of $O(nk)$ since it involves calculating all elements of $\tilde{X}$. The time complexity is also $O(nk)$ for the case of inducing the division of red vertices. The BRIM algorithm will end after a certain number of steps, which in practice few steps are needed since the initial partition is much stable and better than that obtained by random assignment or label-propagation. As a result, the complexity of BRIM algorithm is $O(nk)$, and iterating k from 2 to K brings a time complexity of $O(nK^2)$.

Therefore, the total complexity is $O(pqK + nK^2e + nK^2)$, which is roughly $O(n^3)$ in the worst case. However, in many real networks, the red vertices and blue vertices are often unbalanced. Besides, the subadditivity of matrix rank also shows that $\mathrm{rank}(\mathcal{B}_Q) \le \mathrm{rank}(B) + \mathrm{rank}(d_i g_j^T)$ [29], where the latter is 1 and the rank of $\mathcal{B}_Q$ is usually much smaller than p and q for some networks. In this case, the complexity becomes roughly $O(n^2)$, as we will see in the next section.

4. Experimental results

In this section, we show the performances of the new method on both synthetic and real-world bipartite networks. For synthetic networks, we adopt the *normalized mutual information*(NMI) [30], a measure derived from information theory. By comparing ground truth community structure with the one revealed by the new method, NMI gives a score between 0 and 1. If obtained community partition is identical to the ground truth, the score will take a maximum value of 1. If two partitions are totally independent of each other, the score will be 0. Generally, a higher value of NMI means a more accurate community partition. While for real-world networks without ground underlying partition, Barber's bipartite modularity is used.

4.1. Test on synthetic bipartite networks

To demonstrate the performances of our method, we first conduct experiments on two different synthetic bipartite networks. Besides, We choose the well-known adaptive BRIM and LP&BRIM algorithms to show the differences of distinct strategies followed by BRIM algorithm. We use the implementation of these two algorithms in BiMat, which is a MATLAB library used to analyze biological bipartite networks [31], and our method is implemented in Python. Since the performance of k-means depends on the initial seeds of clusters, we run 10 times for each k in each network and select the best results in the following experiments.

4.1.1. A ring of bipartite cliques

A bipartite clique is a complete bipartite network. If there are M red vertices and N blue vertices in a bipartite clique, then the number of total edges is $M \times N$. It means that each red vertex is adjacent to all N blue vertices, and each blue vertex connects to all M red vertices. To compare the accuracy of different methods, rings of bipartite cliques with different clique numbers are designed. A bipartite network consisting of 4 bipartite cliques with inter-edges from head to tail is shown in Fig. 1(a). In this experiment, we use the bipartite clique with two red vertices and three blue vertices. It has eight edges including six intra-edges and two inter-edges connecting to adjacent cliques: one edge connects one of its red vertices to a blue vertex of an adjacent clique, and the other connects one of its blue vertices to a red vertex of the other clique. The basic topological information of different networks is listed in Table 1, and experimental results are summarized in Table 2.

It is shown in Table 2 that all three methods can perfectly reveal the underlying community structure when there are 4 or 8 bipartite cliques. When the number of bipartite cliques increases to 16, adaptive BRIM gets a partition of 15 communities and LP&BRIM obtains a result with 14 communities. Since our method iterates over different cluster numbers k, it is easy to see in Fig. 2 that when k is larger than the number of ground-truth partition, the method converges to a stable result which is the exact underlying community structure. Besides, the largest modularity obtained by our method

Table 1
Basic topological information of different rings of bipartite cliques.

Network	Red vertices	Blue vertices	Total vertices	Total edges
4 bipartite cliques	8	12	20	28
8 bipartite cliques	16	24	40	56
16 bipartite cliques	32	48	80	112
64 bipartite cliques	128	192	320	448
128 bipartite cliques	256	384	640	896

Table 2
Performances of different methods on rings of bipartite cliques. n_c is the number of communities, NMI measures the accuracy of discovered community partition compared with ground truth and Q_b indicates the bipartite modularity of community partition.

Network	Our method			Adaptive BRIM			LP&BRIM		
	n_c	NMI	Q_b	n_c	NMI	Q_b	n_c	NMI	Q_b
4 bipartite cliques	4	1.000	0.607	4	1.000	0.607	4	1.000	0.607
8 bipartite cliques	8	1.000	0.732	8	1.000	0.732	8	1.000	0.732
16 bipartite cliques	16	1.000	0.795	15	0.984	0.796	14	0.968	0.797
64 bipartite cliques	64	1.000	0.842	62	0.995	0.843	61	0.992	0.845
128 bipartite cliques	128	1.000	0.849	127	0.999	0.850	117	0.988	0.860

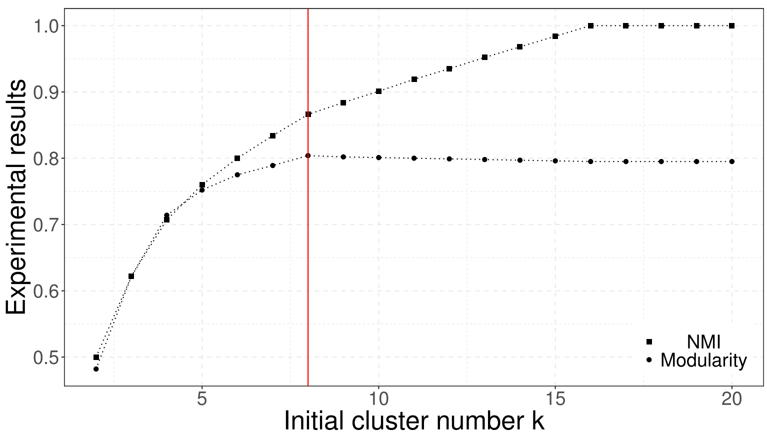

Fig. 2. NMI and Modularity versus k for the ring of 16 bipartite cliques. When k is larger than 16, the results obtained converge to the perfect underlying community structure where NMI is 1. The red vertical line indicates that, the largest bipartite modularity 0.804 appears when k is 8 . (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

is 0.804 which is larger than the results of adaptive BRIM and LP&BRIM. When the number of bipartite cliques continues to increase to 64 or 128, the situation is analogous. Therefore, the bipartite modularity still suffers from the resolution limit like unipartite modularity, which makes it sometimes hard to detect small bipartite cliques. However, our method is helpful to alleviate the problem to some extent. Based on the results of singular value decomposition, the new method can efficiently detect different community partitions for different cluster numbers k. This will give us a more detailed and comprehensive view of community structure. When k is larger than the underlying community number, bipartite modularity decreases monotonically as k increases. The convergence of method ensures a stable community partition, which is the perfect ground-truth in this synthetic networks.

In order to test more thoroughly the complexity of our method, we conducted more experiments on larger rings of bipartite cliques. It is easy to see in Fig. 3, the running time is approximately proportional to the square of the number of vertices.

4.1.2. A ring of unequal sized bipartite cliques

We further design a series of rings with unequal sized bipartite cliques. The minimum number of red vertices or blue vertices is set to two and the maximum number is five, which means the size of bipartite clique is from four to ten. One ring of 4 bipartite cliques is shown in Fig. 1(b). The basic topological information of different rings of unequal sized bipartite cliques is shown in Table 3, where C_{min} and C_{max} indicate the minimum and maximum size of cliques in the network, and further experimental results are summarized in Table 4.

It is clear to see that all three methods can perfectly reveal the underlying community structure when there are 4 or 8 bipartite cliques. When the number of bipartite cliques increases to 16, our method and adaptive BRIM still can get the true community partition, while LP&BRIM obtains a result with 15 communities. Since the number of red and blue vertices in

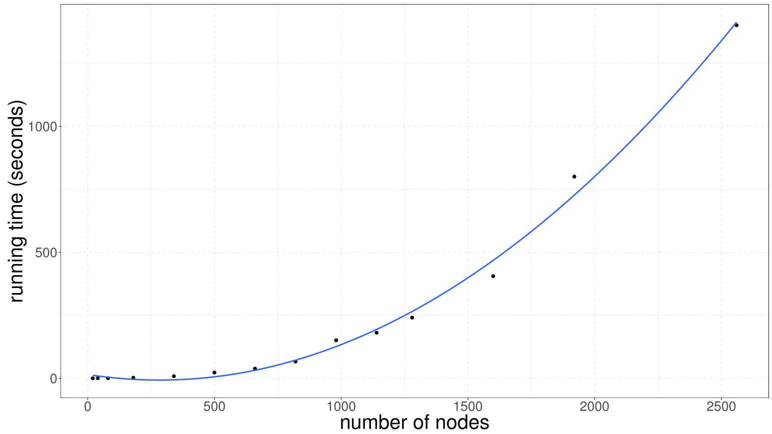

Fig. 3. The running time of our method on different rings of equal sized bipartite cliques, and blue line is fitted by the formula of $y \sim x^2$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 3
Basic topological information of different rings of unequal sized bipartite cliques.

Network	C_{min}	C_{max}	Red vertices	Blue vertices	Total vertices	Total edges
4 bipartite cliques	6	9	12	16	28	50
8 bipartite cliques	5	10	33	24	57	105
16 bipartite cliques	4	10	58	57	115	227
64 bipartite cliques	4	10	212	227	439	836
128 bipartite cliques	4	10	431	445	876	1603

Table 4
Performances of different methods on rings of unequal sized bipartite cliques.

Network	Our method			Adaptive BRIM			LP&BRIM		
	n_c	NMI	Q_b	n_c	NMI	Q_b	n_c	NMI	Q_b
4 bipartite cliques	4	1.000	0.630	4	1.000	0.630	4	1.000	0.630
8 bipartite cliques	8	1.000	0.774	8	1.000	0.774	8	1.000	0.774
16 bipartite cliques	16	1.000	0.855	16	1.000	0.855	15	0.989	0.857
64 bipartite cliques	64	1.000	0.904	62	0.996	0.906	60	0.991	0.908
128 bipartite cliques	128	1.000	0.911	123	0.995	0.913	122	0.995	0.915

each clique is randomly assigned from two to five, the network is more complex than that in former subsection. The changes of NMI and bipartite modularity when k varies for the ring of 64 bipartite cliques are shown in Fig. 4. There are many local optimums of modularity, and the largest modularity detected by our method is 0.923 when k is 37, which is much larger than the results of adaptive BRIM and LP&BRIM. The same as former subsection, when k is larger than 64 which is the number of ground truth partition, new method also converges to a stable partition which is the exact underlying community structure. The situation is analogous when the number of bipartite cliques increases to 128.

4.2. Test on real-world bipartite networks

In this section, we test the new method on two real-world bipartite networks without priori known community structures, and results are evaluated by Barber's bipartite modularity. One network shows the interactions between people and social events, and the other indicates the interlocks among firms in Scotland. These two data sets are widely used for community detection in bipartite network, and generally recognized as benchmarks to compare performances of different methods. Besides, both networks are publicly available.

4.2.1. Southern women event participation

The Southern Women data set was collected by Davis et al. in the town of Natchez, Mississippi, during the 1930s for an extensive study of class and race in black and white society in the Deep South [32]. The network describes the participation of 18 women in 14 social events. The women and social events naturally constitute a bipartite network. Each vertex in this network represents a woman or an event, and an edge connects a woman to an event if the woman participates in the event. Due to its bipartite community structure and small size, this connected and unweighed bipartite network has been extensively studied by social and network scientists.

We let the initial cluster number k range from 2 to 6. The experimental results and comparison with several existing algorithms are summarized in Table 5. It is shown that when k is larger than 4, our method converges to a stable community

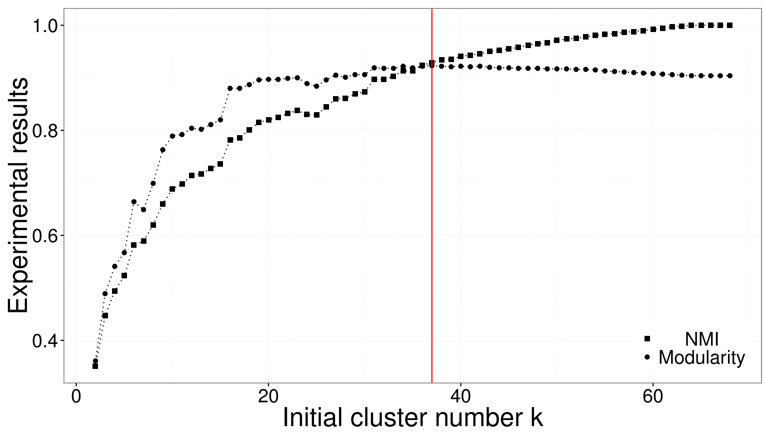

Fig. 4. NMI and Modularity versus k for the ring of 64 unequal sized bipartite cliques. When k is larger than 64, the results obtained converge to the perfect ground truth. The red vertical line indicates that, the largest bipartite modularity is 0.923 when k is 37 . (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 5
Comparison of experimental results in the Southern Women network.

Reference	Method	Maximal modularity
	$k = 2$	0.32117
Ours	$k = 3$	0.34516
	$k = 4, 5, 6$	**0.34554**
Baber	Adaptive BRIM	**0.34554**
Liu & Murata	LP&BRIM	0.32117
	Spectral	0.32117
	Davis 1	0.31057
Barber[a]	Davis 2	0.31839
	Doreian	0.29390
	Unipartite	0.21866
Zhou & Feng[b]	Bi-Louvain	0.32433

[a]These 5 values are reported by Barber in [18].
[b]This value is reported by Zhou and Feng in [21].

partition with 4 communities which is exactly the result obtained by adaptive BRIM. Interestingly, we get a community partition with a modularity of 0.34516 when k is 3 and this community partition is shown in Fig. 5. We get exactly same partition result as LP&BRIM when k is 2. Besides, compared with the results of several existing methods, such as unweighed unipartite projection method, spectral bipartition method [17] and Bi-Louvain method [21], the community assignment of Davis and the network division proposed by Doreian et al. [33], the new method yields a much better community partition. Overall, in the Southern Women network, our method can not only get the same results as adaptive BRIM and LP&BRIM, but also reveal an overlooked community structure when cluster number is 3.

4.2.2. Scotland corporate interlocks

This data set shows the corporate interlocks in Scotland in the beginning of twentieth century(1904–5). It contains 136 multiple directors of 108 Scottish firms [34]. There is an edge linking a director to a firm if the director is one of board members of the firm. This bipartite network is also unweighed, the same as Southern Women network. However, this bipartite network is not fully connected. Therefore, we extract the largest connected component as Barber did in [18], which contains 131 directors and 86 firms. In the following, we only consider the largest subnetwork to which our new method will be applied.

When we vary the initial cluster number k from 2 to the largest possible community number 86, the change of modularity is shown in Fig. 6. The largest bipartite modularity 0.6962 appears when k is 11, and communities are plotted in Fig. 7. The comparison of values of maximal modularity obtained by several existing methods is listed in Table 6. There are many local optimums of modularity in this network, it is difficult for label-propagation or random assignment to find good initializations for BRIM algorithm. We run 30 times for adaptive BRIM and LP&BRIM, the largest modularity obtained by adaptive BRIM is 0.6677 with 17 communities and a partition of modularity 0.6788 with 20 communities is revealed by LP&BRIM. Compared with several existing methods, such as LPb which is a modified label-propagation method for bipartite network, LPAr which has additional randomization when selecting community labels, Hybrid which combines standard LPA and a modularity specialized LPA, and Bi-Louvain which extends Louvain algorithm to bipartite networks, our new method obtains the largest value of bipartite modularity among all these methods. Although Zhan et al. [35] gets a partition with the maximal modularity

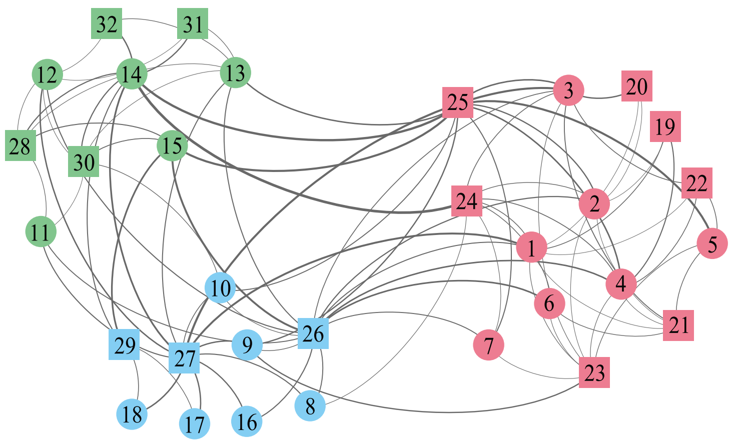

Fig. 5. Community partition in the Southern Women network when k is 3, and the bipartite modularity is 0.34516.

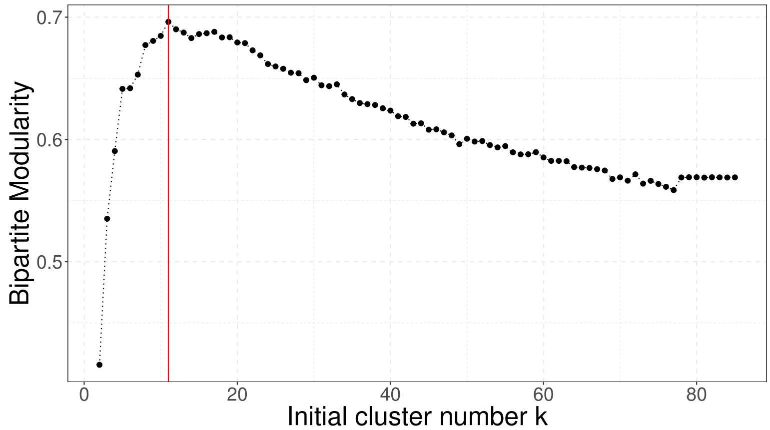

Fig. 6. Bipartite modularity versus k for the Scotland Corporate Interlocks network. The red vertical line indicates that, the largest bipartite modularity appears which is 0.6962 when k is 11 . (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 6
Comparison of values of maximal modularity in the Scotland Corporate Interlocks network.

Reference	Method	Maximal modularity
	Our method	**0.6962**
Baber	Adaptive BRIM	0.6677
Liu & Murata	LP&BRIM	0.6788
	LPA	0.5782
Barber & Clark[a]	LPAb	0.5783
	LPAr	0.6552
	Hybrid	0.5975
Zhou & Feng[b]	Bi-Louvain	0.6930

[a]These 4 values are reported by Barber and Clark in [19].
[b]This value is reported by Zhou and Feng in [21].

of 0.7093 using their MAGA algorithm which is an adaptive genetic algorithm, our method is more intuitive and the result is also comparable in this data set. In summary, our new method gets a community partition of relatively high quality in this network.

5. Conclusion

In this paper, we give a precise analytical expression to reformulate optimizing Barber's bipartite modularity as a spectral problem for a fixed k by relaxing the discreteness condition of community assignment. Besides, we propose a new

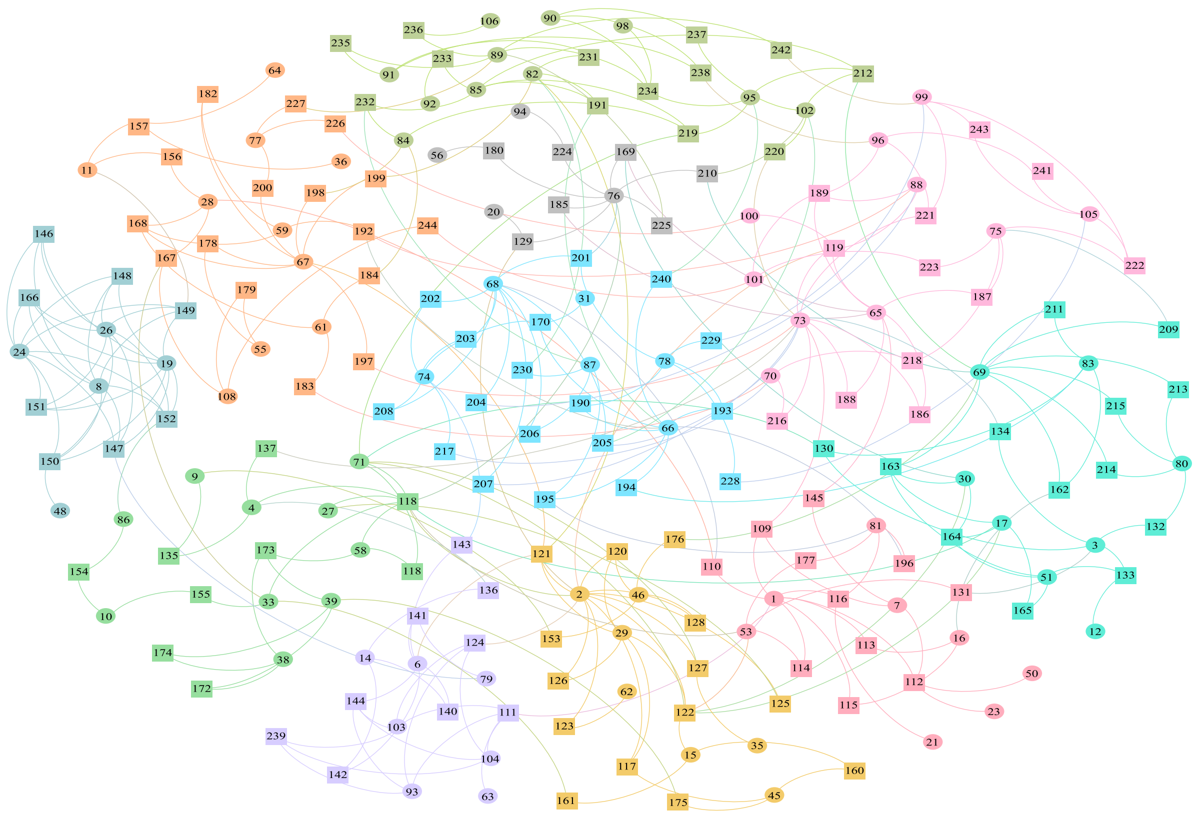

Fig. 7. Community partition found in the Scotland Corporate network.

method combining singular value decomposition and BRIM algorithm to optimize bipartite modularity. Compared with adaptive BRIM and LP&BRIM, the new method adopts the results of k-means on the vector representations of singular value decomposition to initialize the following BRIM algorithm. Though random initialization is simple, it needs to run many times to get a relatively good result. While label-propagation is often stuck in several specific cluster results, which also makes it hard to find the global optimal for complex networks. But for the new method, based on the vector representations of bipartite network vertices, one can easily use fast vector-based cluster methods such as k-means to get relatively good initial partition for each cluster number k, which makes it can efficiently reveal different community partitions of original network. In real-world applications, one can extract a community partition according to a pre-set cluster number or the value of bipartite modularity.

Finally, the new method is carried out in both synthetic and real-world networks. The experimental results also show that, our method is capable to reveal a community partition with larger bipartite modularity than many other existing methods. Besides, it can perfectly converge to the ground truth community structure when k is appropriately set, which helps to alleviate the resolution limit issue. In summary, the new method proposes a more accurate initialization for BRIM algorithm and provides a more comprehensive view of different community structures of bipartite networks.

Acknowledgments

This work is supported by National Natural Science Foundation of China under Grant No. 61425027, Key R&D Project of China under Grant No. 2016YFB0901901, 2017YFC0704100, the 111 International Collaboration Program of China under Grant B06002, and in part by the Special Fund of Suzhou-Tsinghua Innovation Leading Action under Project 2016SZ0202, and Innovation Joint Research Center for Cyber-Physical-Society System. Cangqi Zhou received additional funding from the Natural Science Foundation of Jiangsu Province of China (Grant no. BK20180463).

Appendix A. The relationship between SVD and unipartite projection

Considering a bipartite network $\mathcal{H}$ of m red vertices and n blue vertices where $m < n$, the bipartite adjacent matrix is a $m \times n$ matrix B. According to results of singular value decomposition, the bipartite modularity matrix $\mathcal{B}_Q$ can be approximated as

$$\mathcal{B}_Q \approx U_k \Sigma_k V_k^T$$

where $\mathcal{B}_Q = \mathcal{D} - \mathcal{B}$. If we multiply $\mathcal{B}_Q$ by $\mathcal{B}_Q^T$, we can obtain

$$\mathcal{B}_Q \mathcal{B}_Q^T \approx U_k \Sigma_k^2 U_k^T$$

If we first project the system onto a weighted unipartite network consisted of m red vertices. The adjacent matrix of resulting network is

$$W = BB^T$$

Based on the properties of symmetric matrix, W can be approximated as

$$W \approx P_k \Sigma_{k,w} P_k^T$$

where P_k is the matrix of eigenvectors, and $\Sigma_{k,w}$ is a diagonal matrix whose non-zero elements are the corresponding largest k eigenvalues. Therefore, the difference between the clusters found by SVD and those obtained by unipartite projection is that, U_k and P_k are orthogonal eigenvectors of different matrixes $\mathcal{B}_Q \mathcal{B}_Q^T$ and BB^T.

Appendix B. The flowchart of our method

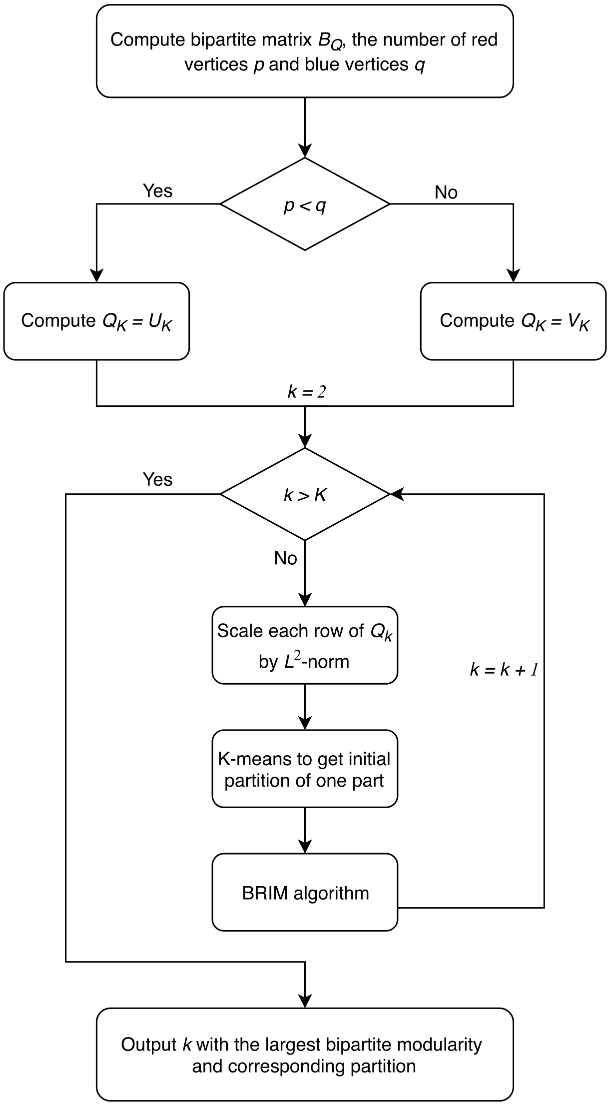

Fig. B.8. Flowchart of our method.

References

[1] M. Girvan, M.E. Newman, Community structure in social and biological networks, Proc. Natl. Acad. Sci. 99 (12) (2002) 7821–7826.
[2] S. Fortunato, D. Hric, Community detection in networks: A user guide, Phys. Rep. Rev. Sect. Phys. Lett. 659 (2016) 1–44, http://dx.doi.org/10.1016/j.physrep.2016.09.002, URL <GotoISI>://WOS:000388062400001.
[3] P. Bedi, C. Sharma, Community detection in social networks, Wiley Interdiscip. Rev. Data Mining Knowl. Discov. 6 (3) (2016) 115–135.
[4] U. von Luxburg, A tutorial on spectral clustering, Stat. Comput. 17 (4) (2007) 395–416, http://dx.doi.org/10.1007/s11222-007-9033-z, URL <GotoISI>://WOS:000249409100009.
[5] T. Heimo, G. Tibély, J. Saramäki, K. Kaski, J. Kertész, Spectral methods and cluster structure in correlation-based networks, Physica A 387 (23) (2008) 5930–5945.
[6] M.E. Newman, M. Girvan, Finding and evaluating community structure in networks, Phys. Rev. E 69 (2) (2004) 026113.
[7] A. Clauset, M.E. Newman, C. Moore, Finding community structure in very large networks, Phys. Rev. E 70 (6) (2004) 066111.
[8] V.D. Blondel, J.-L. Guillaume, R. Lambiotte, E. Lefebvre, Fast unfolding of communities in large networks, J. Stat. Mech. Theory Exp. 2008 (10) (2008) P10008.
[9] P.J. Mucha, T. Richardson, K. Macon, M.A. Porter, J.-P. Onnela, Community structure in time-dependent, multiscale, and multiplex networks, Science 328 (5980) (2010) 876–878, http://dx.doi.org/10.1126/science.1184819, URL <GotoISI>://WOS:000277618800043.
[10] C. Mingming, K. Kuzmin, B.K. Szymanski, Community detection via maximization of modularity and its variants, IEEE Trans. Comput. Soc. Syst. 1 (1) (2014) 46–65, http://dx.doi.org/10.1109/tcss.2014.2307458, URL <GotoISI>://INSPEC:14416087.
[11] J. Shang, L. Liu, X. Li, F. Xie, C. Wu, Targeted revision: A learning-based approach for incremental community detection in dynamic networks, Physica A 443 (2016) 70–85.
[12] S. White, P. Smyth, A spectral clustering approach to finding communities in graphs, in: Proceedings of the Fifth Siam International Conference on Data Mining, Siam, Philadelphia, 2005. URL <GotoISI>://WOS:000289491000025.
[13] M.E. Newman, Finding community structure in networks using the eigenvectors of matrices, Phys. Rev. E 74 (3) (2006) 036104.
[14] J. Ruan, W. Zhang, An efficient spectral algorithm for network community discovery and its applications to biological and social networks, in: Seventh IEEE International Conference on Data Mining, 2007, pp. 643–648.
[15] A. Lancichinetti, S. Fortunato, Limits of modularity maximization in community detection, Phys. Rev. E 84 (6) (2011) 066122.
[16] K.R. Žalik, B. Žalik, A local multiresolution algorithm for detecting communities of unbalanced structures, Physica A 407 (2014) 380–393.
[17] I.S. Dhillon, Co-clustering documents and words using bipartite spectral graph partitioning, KDD-2001, in: Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2001, pp. 269–274. URL <GotoISI>://INSPEC:7306475.
[18] M.J. Barber, Modularity and community detection in bipartite networks, Phys. Rev. E 76 (6) (2007) 066102.
[19] M.J. Barber, J.W. Clark, Detecting network communities by propagating labels under constraints, Phys. Rev. E 80 (2) (2009) 026129.
[20] X. Liu, T. Murata, Community detection in large-scale bipartite networks, Inform. Media Technol. 5 (1) (2010) 184–192.
[21] C. Zhou, L. Feng, Q. Zhao, A novel community detection method in bipartite networks, Physica A 492 (2018) 1679–1693.
[22] R. Guimerà, M. Sales-Pardo, L.A.N. Amaral, Module identification in bipartite and directed networks, Phys. Rev. E 76 (2007) 036102, http://dx.doi.org/10.1103/PhysRevE.76.036102.
[23] A. Mukherjee, M. Choudhury, N. Ganguly, Understanding how both the partitions of a bipartite network affect its one-mode projection, Physica A 390 (20) (2011) 3602–3607.
[24] G.H. Golub, C. Reinsch, Singular value decomposition and least squares solutions, Numer. Math. 14 (5) (1970) 403–420.
[25] M. Brand, Fast low-rank modifications of the thin singular value decomposition, Linear Algebra Appl. 415 (1) (2006) 20–30, http://dx.doi.org/10.1016/j.laa.2005.07.021, URL <GotoISI>://WOS:000236842500003.
[26] A. Frieze, R. Kannan, S. Vempala, Fast monte-carlo algorithms for finding low-rank approximations, J. ACM 51 (6) (2004) 1025–1041, http://dx.doi.org/10.1145/1039488.1039494, URL <GotoISI>://WOS:000226078100006.
[27] C. Elkan, Using the triangle inequality to accelerate k-means, in: Twentieth International Conference on International Conference on Machine Learning, pp. 147–153.
[28] M. Shindler, A. Wong, A. Meyerson, Fast and accurate k -means for large datasets, in: International Conference on Neural Information Processing Systems, pp. 2375–2383.
[29] M. Brookes, The matrix reference manual, in [online] www.ee.ic.ac.uk/hp/staff/dmb/matrix/intro.html.
[30] L. Danon, A. Dí azguilera, J. Duch, A. Arenas, Comparing community structure identification, J. Stat. Mech. 2005 (09) (2005) 09008.
[31] C.O. Flores, T. Poisot, S. Valverde, J.S. Weitz, Bimat: a matlab package to facilitate the analysis of bipartite networks, Methods Ecol. Evol. 7 (1) (2016) 127–132.
[32] A. Davis, B.B. Gardner, M.R. Gardner, W.L. Warner, Deep South: A Sociological Anthropological Study of Caste and Class, University of Chicago Press, 1941.
[33] P. Doreian, V. Batagelj, A. Ferligoj, Generalized blockmodeling of two-mode network data, Soc. Netw. 26 (1) (2004) 29–53.
[34] J. Scott, M. Hughes, The Anatomy of Scottish Capital: Scottish Companies and Scottish Capital, 1900–1979, McGill-Queen's Press-MQUP, 1980.
[35] W. Zhan, Z. Zhang, J. Guan, S. Zhou, Evolutionary method for finding communities in bipartite networks, Phys. Rev. E 83 (6) (2011) 066120.