Scientia Iranica D (2023) 30(3), 1068–1084

Sharif University of Technology
Scientia Iranica
Transactions D: Computer Science & Engineering and Electrical Engineering
http://scientiairanica.sharif.edu

Hybrid adaptive modularized tri-factor non-negative matrix factorization for community detection in complex networks

M. Ghadirian and N. Bigdeli*

Department of Control Engineering, Faculty of Technical and Engineering, Imam-Khomeini International University, Qazvin, Iran.

Received 16 April 2022; received in revised form 7 October 2022; accepted 14 November 2022

KEYWORDS
Community detection;
Tuning parameter;
Non-negative matrix factorization;
Modularized regularization;
Singular value decomposition.

Abstract. Community detection is a significant issue in extracting valuable information and understanding complex network structures. Non-negative Matrix Factorization (NMF) methods are the most remarkable topics in community detection. The Modularized tri-factor NMF (Mtrinmf) method was proposed as a new class of NMF methods that combines the modularized information with tri-factor NMF. It had high computational complexity due to its dependence on the choice of the initial value of its parameter and the number of communities (c). In other words, the Mtrinmf method should search among different candidates to find correct c. In this paper, a novel Hybrid adaptive Mtrinmf (Hamtrinmf) method is proposed to improve the performance of Mtrinmf and reduce the computational complexity efficiently. In the proposed method, computational complexity reduction is made possible by selecting the right c candidates and tuning parameter. For this purpose, a hybrid algorithm including Singular Value Decomposition (SVD) and Relative Eigenvalue Gap (REG) algorithms is suggested to estimate the set of c candidates. Next, the Tuning parameter Mtrinmf (Tpmtrinmf) model is proposed to improve the performance of community detection via employing a self-tuning β parameter. Moreover, experimental results confirm the efficiency of the Hamtrinmf method with respect to other reference methods on artificial and real-world networks.

1. Introduction

Interconnected data and systems of different areas can be modeled as complex networks. These networks represent community structures while understanding these communities can provide helpful information about interconnections like determining the closest profiles in social networks [1]. Therefore, community detection is an important problem in complex network analysis, which clusters complex networks into cohesive subgroups [2]. In recent years, the community detection algorithm has been used in various scientific fields, such as nonlinear system decomposition and feedback control design in electrical engineering [3], entity resolution and profile matching in social networks analysis [4], as well as exploring and understanding complex network structures like biological networks, social networks, and citation networks [1]. Nowadays, substantial attention has been drawn to community detection, and this topic has been presented based on many methods from different points of view. Some of these methods are single-objective and multi-objective evolutionary optimization [5,6], fuzzy clustering [7], La-

*. *Corresponding author. Tel.: +98 2833901296;*
E-mail address: n.bigdeli@eng.ikiu.ac.ir (N. Bigdeli)

doi: 10.24200/sci.2022.59919.6703

bel Propagation Algorithm (LPA) [8,9], random walk [10], and Non-negative Matrix Factorization (NMF).

Community detection based on NMF is a powerful method for clustering data. Generally, the NMF algorithm has found extensive applications in various fields such as audio source separation in audio mining [11], image processing [12], prominent topics contained in a document corpus in text mining [13], and community detection in graph mining [14]. These fields include networks clustering with different similarity measures [15], clustering improvement using topology and structure similarity information [16], fuzzy clustering [17], semi-supervised community detection method using prior information [18,19], robust NMF model using noisy prior information [20], as well as multi-view and adaptive clustering using the link and content information [14]. Moreover, community detection based on NMF is a robust tool for detecting hidden communities in various networks [21–23].

However, the NMF-based community detection methods suffer from high computational complexity or low accuracy of clustering by increasing the size of networks due to the need to tune the internal hyperparameters and to determine the number of communities (c) [14,24]. Therefore, they are not scalable or proper for large networks. Actually, one of the important requirements in NMF models is to select c. in advance. Over the years, several approaches have been presented to determine c. Some sequential approaches including maximum modularity [24], maximum modularity densities [25], and maximum general modularity densities [25] aim to find the best c. On the other hand, some approaches estimate c according to network features, such as Singular Value Decomposition (SVD) [26,27], cross-validation [28], Classical Multi-Dimensional Scaling (CMDS), and Relative Eigenvalue Gap (REG) [29]. Obtaining the best c candidates from measurements of complex network structures is of higher accuracy. However, it is more time-consuming than estimation approaches. While estimating c via less computational methods is preferred, higher accuracy relates to network features with a long computation time. Thus, identifying c is still an open challenging issue for many networks, as emphasized in the related literature [25,26]. Therefore, in this paper, a new method is developed to estimate the correct c to be used in NMF models.

As reviewed, the NMF-based community detection is a general method for all data types; therefore, it does not consider any network feature and hidden topological information. Modularity is one of the various measurement criteria to evaluate the quality of graph partitioning [1]. This criterion considers the strength of relationship density of each edge within every community and can regard the related nodes between communities [30]. For this reason, the modularity criterion as an important criterion of network clustering has been combined with the NMF method to make the most of network feature [25,31]. Specifically, Modularized tri-factor Non-negative Matrix Factorization (Mtrinmf) was introduced in [25] as an advanced NMF method for community detection. In the Mtrinmf method, the modularized information is combined with the NMF as the regularization term to improve the performance of community detection. The method caused a significant improvement in identifying partitions and modularity information. As its drawback, the Mtrinmf model has a constant β, serving as a tuning parameter whose best value is selected by trial and error. The tuning β parameter via the current trial and error strategy imposes significant computational complexity on this method. In addition, as a general drawback of the NMF-based methods, selecting the best number of communities (c) is a challenge that increases the computational complexity of Mtrinmf.

Regarding the above discussions, in order to improve performance and reduce the computational complexity of the Mtrinmf, in this paper, a new algorithm is proposed to obtain the number of clusters and choose the best value for the tuning parameter β. As a result, the hybrid of SVD and REG algorithms will be adopted to determine the set of c candidates with length K to reduce the dependence on network features. The SVD algorithm is used to compute singular values of the adjacent matrix, and the REG algorithm determines the value of K and then estimates the set of c candidates with length K using the output of the SVD algorithm.

Secondly, a novel NMF model called Tuning parameter modularized tri-factor NMF (Tpmtrinmf) will be proposed, which improves the performance of community detection using a self-tuning β parameter. Afterward, the best community candidates are selected based on general modularity density information.

These steps form a promising novel community detection method named Hybrid adaptive Modularized tri-factor NMF (Hamtrinmf). By employing the proposed method, the computational complexity is reduced compared to the Mtrinmf method via selecting the right c candidates and tuning the parameters. Therefore, the main achievements of the proposed method are high accuracy, internal adaptivity, low complexity, and scalability for large networks. The viability of the proposed algorithms would be investigated on nine small/large-sized real-world networks and two artificial ones.

In summary, the main achievements of this paper are as follows:

- Unlike the Mtrinmf model, the Tpmtrinmf model improves performance using a self-tuning β parameter, and an iterative method is developed for

tuning β. This achievement not only reduces the computational volume by avoiding the trial and error in tuning β, but also improves the performance of the community detection model;

- The provision of a solution for selecting the set of c candidates is one of the innovations of this paper, which reduces the computational volume, improves performance of the community detection, and is robust against the graph features;
- The proposed Hamtrinmf method outperforms other reference methods by applying the set of c candidates with length K by using the Tpmtrinmf model and best community selector. Moreover, it has reduced computational complexity compared to the Mtrinmf method;
- The proposed approach has been evaluated on various artificial and real-world networks. According to the experimental results, it is demonstrated that the Hamtrinmf and Tpmtrinmf models outperform other compared methods including the Mtrinmf model.

The rest of the paper is organized as follows: Section 2 reviews related works on NMF algorithms. Section 3 demonstrates the proposed method and analyzes its computational complexity. Section 4 presents several experimental results of comparing the proposed method with the other representative methods. Finally, Section 5 summarizes the proposed procedures and achievements.

2. Related works

According to the previous section, community detection based on the NMF methods includes a class of efficient methods for clustering complex networks. However, these methods have some disadvantages compared to other community detection methods, such as dependency on choosing the best internal parameters and the need for pre-determining the number of communities (c). Therefore, in this section, NMF-based community detection methods are reviewed and discussed first. Then, several approaches such as modularity and general modularity density indices are presented to estimate correct c values. Finally, extended models based on NMF are reviewed at the end of this section.

2.1. NMF-based community detection algorithms

Given an adjacent matrix $A \in R^{n\times n}$ for the input graph with n nodes and m edges, NMF models factorize a given similarity matrix A into two new matrices $W \in R^{n\times c}$ and $H \in R^{n\times c}$: $A \simeq WH^T$ where W and H are called community indicator feature matrix and community relation matrix, respectively. The error between A and WH^T is measured by the cost function $J_{NMF}(W,H)$. W and H are found by minimizing $J_{NMF}(W,H)$ as follows:

$$\min_{W,H} J_{NMF}(W,H) = ||A - WH^T||_F^2, \quad (1)$$

where $||.||_f$ stands for the Frobenius norm. As an extension to NMF, Symmetric Nonnegative Matrix Factorization (SNMF) can drastically improve community detection. In SNMF, A is assumed to be a symmetric matrix and the objective function would be rewritten as follows [25]:

$$\min_{H} J_{SNMF}(H) = ||A - HH^T||_F^2. \quad (2)$$

One of the other extensions of NMF model is tri-factor NMF (triNMF) model that can factorize matrix A into two non-negative matrices $W \in {R_+}^{c\times c}$ and $H \in {R_+}^{n\times c}$, where $A \simeq HWH^T$ [32,33]. H and W are the community membership and community relation matrix, respectively, and c denotes the number of communities. The error between A and Trinmf model (HWH^T) is measured by cost function $J_{Trinmf}(W,H)$ as follows:

$$\min_{W,H} J_{Trinmf}(W,H) = \left\{ \sum_{i=1}^{n} \sum_{j=1}^{n} \left(A_{ij} - \left(HWH^T\right)_{ij}\right)^2 | \right\},$$

s.t.:

$$H \in R_+^{n\times c}, W \in R_+^{c\times c}, \sum_{r=1}^{c} H_{ir} = 1, \forall i = 1,2,...,n, \quad (3)$$

where H_{ij} and W_{ij} denote the existence probability of node i in community r and the existing probability of edges between i and communities j, respectively.

2.2. Several approaches to estimating correct c values

2.2.1. Modularity

The modularity criterion determines the validity of community detection based on the density of edges in each cluster and intergroup communication [1,25]. Therefore, in general, community detection based on modularity criteria can be rewritten as an optimization problem with the following conditions [25]:

$$Q = \frac{1}{2m} tr(H^T BH), B = A - B_1, (B_1)_{ij} = \frac{k_i k_j}{2m}, \quad (4)$$

where B is the modularity matrix, A is the adjacent matrix, k_i is the degree corresponding to the nth node, $X = [X_{ij}j] \in R^{n\times c}$ is the members matrix for each cluster, and $(B_1)_{ij} = \frac{k_i k_j}{2m}$.

2.2.2. General modularity density

The general modularity density index considers the average inner degree and outer degree of each cluster. The inner degree refers to the sum of the edges of interval nodes in each cluster, and the outer degree is the sum of edges between the nodes of the cluster with

the nodes of another cluster. It can be rewritten for k number of partitions ($\{V_c\}_{(r=1)}^{c}$) as follows [34]:

$$D_\lambda(\{V_c\}_{r=1}^{c}) = \sum_{r=1}^{c} \frac{2\lambda l(V_r, V_r) - 2(1-\lambda) l(V_r, \overline{V}_r)}{|V_r|}, \quad (5)$$

where:

$$l(V_1, V_2) = \sum_{i \in V_1, j \in V_2} A_{ij}, l(V_1, V_2) = \sum_{i \in V_1, j \in V_2} A_{ij}, \overline{V}_1 = V \backslash V_1,$$

and V_r is the set of vertices in the rth community. Furthermore, D_λ evaluates small and large clusters by using ratio association and ratio cut for $\lambda < 0.5$ and $\lambda > 0.5$, respectively. Therefore, D_λ equals modularity density when $\lambda = 0.5$. Advantages such as selecting the best communities with different sizes, not dividing cliques, and resolving graph types are obtained by selecting different λ values.

It is noteworthy that general Q_D is a generalization of the modularity density, which is a combination of the ratio cut and ratio association [19,34] and can be written in a matrix form as in the following:

$$Q_D = tr(U^T(2A - 2(1-\lambda)C)U), \quad (6)$$

where C is a diagonal matrix with values $C_{ii} = \sum_{j=1}^{n} A_{ij}$, U denotes a column vector whose element is u_r which satisfies $u_r = \frac{s_{ir}}{\sqrt{n_r}}$. Moreover, n_r represents the number of nodes in the rth community and s_{ir} indicates that node i belongs to the rth community. Q_D equals the ratio association, modularity density, and ratio cut for $\lambda = 0$, $\lambda = 0.5$, and $\lambda = 1$, respectively.

2.3. Extended models based on NMF

In recent years, a challenging topic for improving the NMF-based methods has been the use of network features (structure information, topological information, and network specifications) or prior information. These methods have resolved some of the shortcomings, but they are still dependent on their internal parameter and there is a need to select the correct c.

Although the Standard NMF model can offer network partitions, it may ignore some essential network specifications such as modularity information [25]. Yan and Chang [25] indicated Mtrinmf using modularity information. The Mtrinmf model can be rewritten as follows:

$$\min_{W,H} J_{Mrinmf}(W, H) = \left\{ \sum_{i=1}^{n} \sum_{j=1}^{n} (A_{ij} - (HWH^T)_{ij})^2 - \beta tr(H^T B H) \right\}$$

s.t.:

$$H \in R_+^{n \times c}, \quad W \in R_+^{c \times c}, \quad \beta > 0,$$

$$\sum_{r=1}^{c} H_{ir} = 1, \quad \forall i = 1, 2, ..., n, \quad (7)$$

where β is the weight of the modularized regularization. Yan and Chang [25] showed that modularity information enhancement made the Mtrinmf model perform better than Trinmf. In addition, they demonstrated that the Mtrinmf method, which is named Mtrinmf+general Q_D, eliminated the requirement to set the candidate number of communities (c) by employing general Q_D.

Lu et al. [27] proposed two semi-supervised NMF-based methods: SVDCNMF and SVDCSNMF, which used prior information to improve community detection. The SVDCNMF and SVDCSNMF methods improved standard NMF and SNMF models, respectively, and the objective function was represented as:

$$\min_H J_{SVDCSNMF}(H) = \left|\left|A - HH^T\right|\right|_F^2 + 2\lambda tr(H^T L H), \quad (8)$$

where L is the graph Laplacian of prior information. In Eq. (8), $\left|A - HH^T\right|_F^2$ refers to SNMF models and $2\lambda tr(H^T L H)$ represents the prior information term. The SVD algorithm detects c, and the trial-and-error approach estimates the correct λ value.

Wu et al. [16] proposed a mixed hypergraph NMF named Modularized Deep Nonnegative Matrix Factorization (MDNMF) by combining NMF with hypergraph regularization, which encoded the higher-order information into NMF by hypergraph and made use of structure similarity information and topological connection information. The MHGNMF model was divided into $MHGNMF_{kl}$ and $MHGNMF_{sq}$ models based on the type of the community detection function. However, the $MHGNMF_{sq}$ algorithm is preferred due to use of the Frobenius norm in its optimization function. The objective function of $MHGNMF_{sq}$ is defined as:

$$\min_{W,H} J_{MHGNMF}(W, H) = \left|\left|A - WH^T\right|\right|_F^2 + \beta tr(H^T L_h H), \quad (9)$$

where L_h is a hyperlaplacian matrix, and $tr(H^T L_h H)$ term presents the structure similarity and topological connection information. Both algorithms select the best β using the trial-and-error approach.

Additionally, Zhang and Zhou [35] suggested a new model named MDNMF, which combined modularity with DNMF-based community detection. The MDNMF model is composed as follows:

$$\min_{(U_i, H, M, C)} J_{MDNMF}(U_1, ..., U_n, H, M, C)$$

$$= \left|\left|A - U_1 ... U_p H^T\right|\right|_F^2 + \alpha \left|\left|M - H^T C^T\right|\right|_F^2 - \beta tr(H^T B H) + \lambda tr(HLH^T),$$

s.t.:

$$U_i, M, C, H \geq 0, \quad \forall i = 1, 2, ..., p, \tag{10}$$

where L and λ denote the graph Laplacian matrix and the regularization parameter, respectively, and $\lambda tr(HLH^T)$ utilizes a regularized graph. Also, the MDNMF model depends on the choice of α, β, and λ to extract the best cluster.

Ye et al. [36] proposed a novel model, named Deep Autoencoder-like NMF (DANMF), for community detection. Similar to deep autoencoder, DANMF consisted of an encoder component and a decoder component. This model was defined as:

$$\min_{(U_i, H_p)} J_{DANMF}(U, H) = \left|\left|A - U_1 ... U_p {H_p}^T\right|\right|_F^2 + \left|\left|H_p - U_1^T ... U_n^T A\right|\right|_F^2 + \lambda tr(H_p L {H_p}^T),$$

s.t.:

$$U_i, H_p \geq 0, \quad \forall i = 1, 2, ..., p. \tag{11}$$

It should be noted that according to Eq. (11), choosing the best λ parameter would improve the DANMF model. As a result, the performance efficiency of this method depends on the correct selection of the λ parameter. Therefore, eliminating the trial-and-error approach to selecting this parameter may reduce the computational complexity and improve the efficiency of this method, as well.

3. Proposed community detection method

As mentioned earlier, in the present paper, a new Hamtrinmf method will be proposed to improve the performance and reduce the computational complexity of the Mtrinmf method. Figure 1 demonstrates the flowchart of the Hamtrinmf method. According to Figure 1, the Hamtrinmf method is planned in three parts as follows:

- **Determining a set of c candidates with length K**: The SVD algorithm obtains singular values of the adjacent matrix. Furthermore, the REG algorithm identifies the value of K and collects the index of the largest members of the REG set with length K and, then, organizes a set of c candidates;
- **Tpmtrinmf model**: a new model is proposed to enhance the performance of the Mtrinmf model by introducing a self-tuning parameter;
- **Best community selector**: The general modularity density is used to identify the best community detection among the K outputs of the Tpmtrinmf model for complex networks.

In this section, three parts of the Hamtrinmf method are introduced. Afterward, the computational complexities of the Hamtrinmf and Mtrinmf methods are computed to demonstrate that complexity of the Hamtrinmf method is lower than that of the Mtrinmf.

3.1. Determining a set of c candidates with length K

Several methods have been proposed to determine the number of communities (c) [14–16]. In this paper, a hybrid method based on SVD and REG is employed. The SVD equation for adjacent matrix A is expressed as follows:

$$A = USV^T, \ S = diag\{\sigma_1, \sigma_2, ..., \sigma_n\}, \tag{12}$$

where S denotes a diagonal matrix and diagonal entries σ_i of S are singular values of A. After calculating singular values, it is needed to find differences between the two eigenvalues of the matrix called spectral or eigenvalue gap, which is used for determining c. The REG method suggested in [16] was first used in CMDS methods. Since the eigenvalues found by the REG method for CMDS are similar to the singular values of the adjacent matrix A, in this paper, the generalized REG method is extended to identify the existing gaps. The output of the REG algorithm for SVD outputs is as follows:

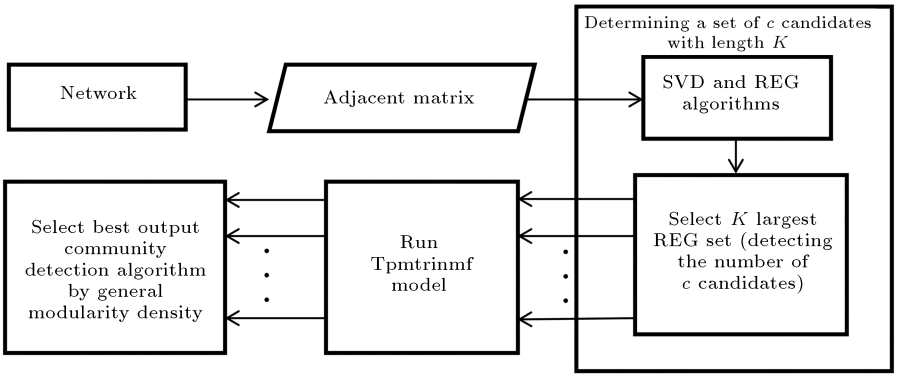

Figure 1. Flowchart of the Hamtrinmf method.

$$REG_i = \frac{\sigma_{i-1} - \sigma_i}{\sigma_i}, \quad \forall i = 2, ..., h,$$

$$REG = \{REG_2, REG_3, ..., REG_h\}, \tag{13}$$

where h indicates the number of positive singular values in the mentioned adjacent matrix; on the other hand, it is better to ignore singular values close to zero because it may cause a computational error.

For determining the set of c candidates with length K, the RS set (Eq. (14)) is formed from sorting REG (Eq. (6)) from maximum to minimum values ($REG_{j_1} > REG_{j_2} > ... > REG_{j_{h-1}}, j_1, j_2, ..., j_{h-1} \in \{2, ..., h\}$).

$$RS = \{REG_{j_1}, REG_{j_2}, ..., REG_{j_{h-1}}\}. \tag{14}$$

Finally, the set of c candidates (Eq. (15)) includes the indexes of the first K members of RS set.

$$c \in \{j_1, j_2, ..., j_k\}. \tag{15}$$

For identifying K values, the gaps between members of RS are computed as performed in the REG algorithm (Eq. (16)):

$$REG_{RS} = \frac{REG_{j_{i-1}} - REG_{j_i}}{REG_{j_i}}, \quad \forall i = 1, ..., h-1,$$

$$REG_{RS} = \{REG_{RS_1}, REG_{RS_2}, ..., REG_{RS_{h-1}}\}. \tag{16}$$

At last, K is equal to the index of the maximum value of REG_{RS} set as follows:

$$K = \arg\max\{i \,|\max\{REG_{RS_i}, i = 1, ..., h-1\}\}. \tag{17}$$

Figure 2 is a toy example for small graphs. The input graph is first converted into an adjacent matrix and, then, for the proposed SVD and REG algorithms, three different diagrams of singular value, REG, and REG_{RS} are plotted with respect to Eqs. (12), (14), and (16). Finally, the best set of c with length K is computed by Eq. (17).

3.2. Tpmtrinmf model

The dependency of the Mtrinmf model on the selection of β parameter [25] prompted the authors to modify the cost function of Mtrinmf to include the self-tuning β parameter to improve its performance. On the other hand, as mentioned earlier, in Eq. (7), $\sum_{i=1}^{n}\sum_{j=1}^{n}(A_{ij} - (HWH^T)_{ij})^2$ is a clustering term and is a modularity term, which is added to improve the performance. Since β has no upper bound and the clustering term is more important, the cost function of the Mtrinmf must always be positive. As a result, the modified cost function including the self-tuning β parameter is designed as the following:

$$\min_{W,H,\beta} J_{Tpmtrinmf}(W, H, \beta) =$$

$$\left\{\sum_{i=1}^{n}\sum_{j=1}^{n}(A_{ij} - (HWH^T)_{ij})^2 - \beta tr(H^T BH)\right\},$$

s.t.:

$$H \in R_+^{n\times c}, W \in R_+^{c\times c}, \beta > 0,$$

$$\sum_{r=1}^{c} H_{ir} = 1, \forall i = 1, 2, ..., n, \tag{18}$$

where A and B matrices are symmetric; therefore, the input graph is undirected. Moreover, W matrix is symmetric, as well.

Since the matrices W and H as well as the parameter β are nonnegative, the cost function in Eq. (18) can be solved by the Lagrangian method. Therefore, the Lagrangian cost function is derived as:

$$\begin{aligned} L(W, H, \beta) = {} & tr(AA^T) - 2tr(AHW^TH^T) \\ & + tr(HWH^THW^TH^T)\Psi \\ & - \beta tr(H^TBH) + tr(\theta H^T) \\ & + tr(\phi W^T) + \psi\beta. \end{aligned} \tag{19}$$

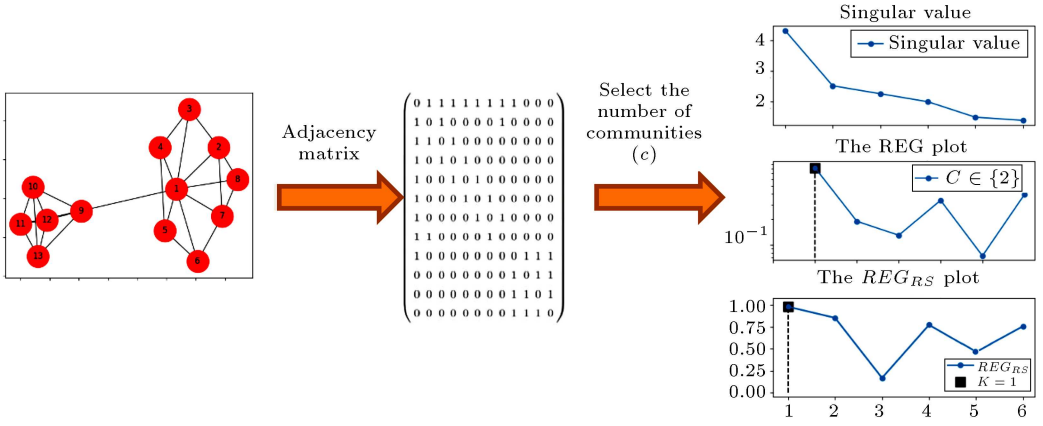

Figure 2. Flowchart of the SVD and REG algorithms.

If θ, ϕ, and ψ are the Lagrangian coefficients for constraints W, H, $\beta \geq 0$; then, derivative of the $L(W, H, \beta)$ function with respect to W, H, and β will be as follows:

$$\frac{\delta L}{\delta H} = \phi + 4HWH^THW^T - 4AHW^T - 2\beta BH,$$

$$\frac{\delta L}{\delta W} = \theta - 2H^T(A - HWH^T)H,$$

$$\frac{\delta L}{\delta \beta} = \psi - tr(H^TBH). \quad (20)$$

Finally, using $B = A - B$ and according to Karush-Kuhn-Tucker (KKT) condition, i.e., $\psi\beta = 0$, $\theta_{ir}h_{ir} = 0$, and $\phi_{rs}w_{rs} = 0$, Eq. (21) will be derived as:

$$(4HWH^THW^T)_{ir}h_{ir} - (4AHW^T)_{ir}h_{ir}$$

$$-(2\beta AH)_{ir}h_{ir} + (2\beta B_1H)_{ir}h_{ir} = 0,$$

$$2(H^THWH^TH)_{rs}w_{rs} - 2(H^TAH)_{rs}w_{rs} = 0,$$

$$tr(H^TB_1H)\beta - tr(H^TAH)\beta = 0. \quad (21)$$

Therefore, iterative updating rules for W, H, and β are expressed as follows:

$$H_{ir} := H_{ir}\frac{(2AHW^T + \beta AH)_{ir}}{(2HWH^THW^T + \beta B_1H)_{ir}},$$

$$W_{rs} := W_{rs}\frac{(H^TAH)_{rs}}{(H^T(HWH^T)H)_{rs}},$$

$$\beta := \beta\frac{tr(H^TAH)}{tr(H^TB_1H)}. \quad (22)$$

In order to establish the condition of $\sum_{r=1}^{c} H_{ir} = 1$ in the optimization function, it is sufficient to have:

$$H_{ir} = \frac{H_{ir}}{\sum_{r=1}^{c} H_{ir}}. \quad (23)$$

3.3. Convergence analysis and calculated upper bound of β

In order to prove the convergence shown in Eq. (22) and (23), the following theorems are considered.

Theorem 1: $J_{Tpmtrinmf}$ in Eq. (18) does not increase by updating H via Eq. (22) while fixing other variables.

Theorem 2: $J_{Tpmtrinmf}$ in Eqs. (18) does not increase by updating W via Eq. (22) while fixing other variables.

Theorem 3: $J_{Tpmtrinmf}$ in Eq. (18) does not increase by updating β via Eq. (22) while fixing other variables.

It is noted that for a brief report, the proof of Theorem 1 was given in [37] and the proofs of Theorems 2 and 3 were presented in [14].

The only thing that can be presented in the proofs of Theorem 1 is the constraint on the β parameter for convergence. If the main function is defined, the proof of the first theorem is as follows:

$$F(H) = -2tr(AHW^TH^T) + tr(HWH^THW^TH^T)$$

$$- \beta tr(H^TBH). \quad (24)$$

The first- and second-order derivatives are as follows:

$$\frac{\delta F}{\delta H_{ij}} = (4HWH^THW^T - 4AHW^T - 2\beta BH)_{ij}, \quad (25)$$

$$\frac{\partial^2 F}{\partial {H_{ii}}^2} = (12HWW^TH^T - 4AW^T - 2\beta B^T)_{ii}. \quad (26)$$

In order for the second-order derivative function to be positive, it is sufficient to define the β parameter as follows:

$$(12HWW^TH^T - 4AW^T - 2\beta B^T)_{ii} >$$

$$0 \rightarrow \beta < \frac{6(HWW^TH^T)_{ii} - 2(AW^T)_{ii}}{B_{ii}}. \quad (27)$$

Therefore, the parameter β is constrained by the following boundaries:

$$0 < \beta < \min\left(\frac{6(HWW^TH^T)_{ii} - 2(AW^T)_{ii}}{B_{ii}}\right). \quad (28)$$

Finally, according to Eqs. (22) and (28), the Tpmtrinmf model is proposed in Algorithm 1.

3.4. Best community selector

In recent years, several model selection methods, such as model selection based on modularity information [38], model selection based on modularity density information [19], and general modularity density information [25], have been employed to recognize the best community selection. According to the advantages of general modularity density information in [25], in the present paper, this criterion was used to select the best community.

Figure 3 is a sample chart presenting the proposed Hamtrinmf algorithm for a toy example. In this method, first, the best candidates of c are found by REG and SVG algorithms and then, several practicable clusters are proposed for Tpmtrinmf model. At last, by using general modularity density, the best clustering is chosen.

4. Experiments, results, and analysis

In this section, various real-world and artificial test networks are introduced first. Then, evaluation matrices are discussed to evaluate the performance and capabilities of the proposed method compared to othercommunity detection approaches. Finally, the computational complexities of the Hamtrinmf and Mtrinmf methods are compared.

Input:
Adjacency matrix A
Number of communities c
Number of iterations I_t

Output:
Clustering label of each node

1: **Initialize** H, W **and** β

2: **For** $t = 1 : I_t$ **do**

3: $H_{ir} \coloneqq H_{ir} \dfrac{(2AHW^T + \beta AH)_{ir}}{(2HWH^T HW^T + \beta B_1 H)_{ir}}$

4. $H_{ir} = \dfrac{H_{ir}}{\sum_{r=1}^{c} H_{ir}}$

5. $W_{rs} \coloneqq W_{rs} \dfrac{(H^T AH)_{rs}}{(H^T (HWH^T)H)_{rs}}$

6. $\beta \coloneqq \beta \dfrac{tr(H^T AH)}{tr(H^T B_1 H)}$

7. If $(\beta > \min(\dfrac{6(HWW^T H^T)_{ii} - 2(AW^T)_{ii}}{B_{ii}}))$ and $(\dfrac{6(HWW^T H^T)_{ii} - 2(AW^T)_{ii}}{B_{ii}} > 0)$:

$$\beta = \frac{6(HWW^T H^T)_{ii} - 2(AW^T)_{ii}}{B_{ii}}$$

8. **End for**

9. **Return** $(v_i, I_i) = \arg\max_{r \le c} H_{ir}$

Algorithm 1. Tpmtrinmf model.

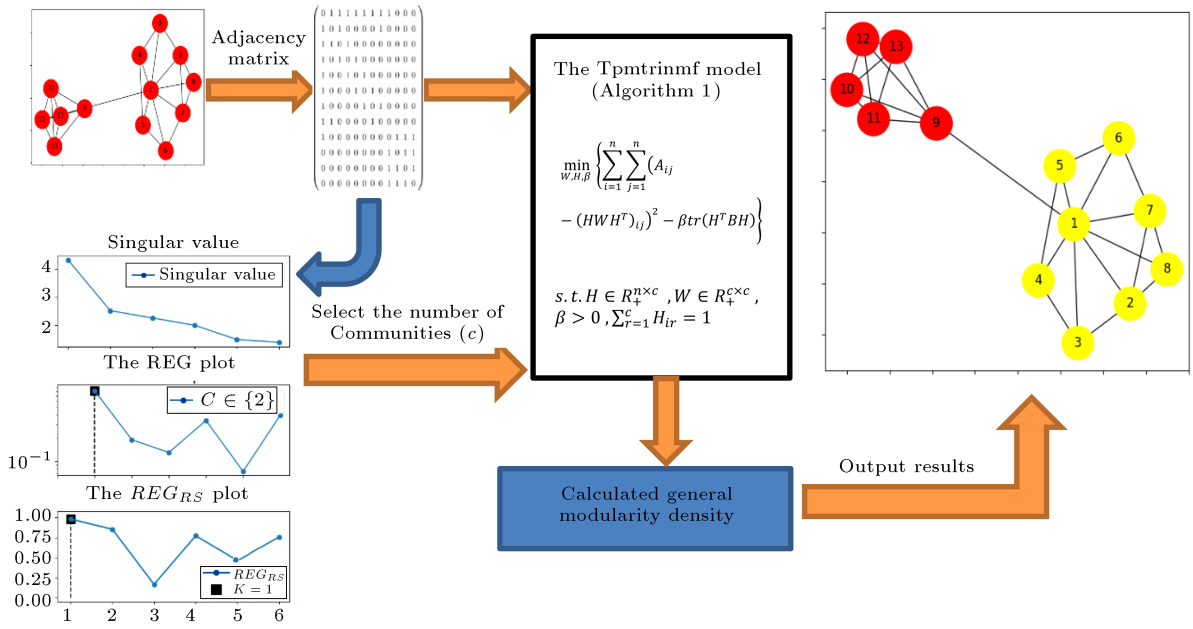

Figure 3. Flowchart of Hamtrinmf method.

4.1. Datasets

In this paper, nine real-world and two artificial benchmark networks are selected and used to evaluate different community detection methods. The real-world datasets include the Zachary karate club network (Karate) [39], Jazz network (Jazz) [40], Political books network (Political books) [41], Lusseau's bottlenose dolphins social network (Dolphins) [42], and American college football network (Football) [43] as small real-world networks, while the Political blogosphere net-

Table 1. Real-world networks information.

Networks	n	m	$\bar{c}$
Karate [39]	34	78	2
Jazz [40]	198	2742	4
Political books [41]	105	441	3
Dolphins [42]	62	159	4
Football [43]	115	613	12
Polblogs [21]	1490	16718	2
Cora [21]	2708	5429	7
Citeseer [21]	3312	4732	6
Pubmed [21]	19717	44338	3

work (Polblogs), Cora citation network (Cora), Citeseer digital library networks (Citeseer), and Pubmed citation network (Pubmed) stand for large real-world networks [21]. The information on real-world networks is listed in Table 1. As mentioned before, $\bar{c}$ denotes the number of ground-truth communities. In addition, two artificial networks named Girvan-Newman (GN) and Lancichinetti-Fortunato-Radicchi (LFR) are presented in the following:

- **LFR:** The LFR networks were introduced by Lancichinetti et al. [44]. This network has some essential characteristics of networks, such as power-law distribution of node degrees and community size. In order to evaluate community detection methods on this network, the parameters of generated LFR network are defined as follows. The number of nodes is 700. The network's average degree and maximum degree are 20 and 50, respectively. The power-law exponent for degree distributions is -3, and the power-law distribution of community size is -1. Note that the community size ranges from 20 to 60 nodes. Finally, the mixing parameter μ varies between 0.1 and 0.9.
- **GN:** The GN network was proposed by Girvan-Newman [25]. This network consists of 128 nodes and four non-overlapping communities, with 32 nodes in each community. The average degree of each node is equal to $Z_{in} + Z_{out} = 16$, where Z_{in} and Z_{out} denote the internal and external degrees of the nodes, respectively.

4.2. Assessment standards

In this paper, Normalized Mutual Information (NMI) and modularity information (Q) are applied to assess the accuracy of different community detection methods. The NMI information is extensively applied in networks to compare the similarity between the ground truth partition labels and partition labels. The NMI information is defined as Eq. (29):

$$NMI(C,C') = \frac{-2\sum_{i=1}^{|C|}\sum_{j=1}^{|C'|} n_{C_i\cap C'_j}\log\left(\frac{n_{C_i\cap C'_j}n}{n_{C_i}n_{C'_j}}\right)}{\sum_{i=1}^{|C|} n_{C_i}\log\left(\frac{n_{C_i}}{n}\right)+\sum_{j=1}^{|C'|} n_{C'_j}\log\left(\frac{n_{C'_j}}{n}\right)}, \tag{29}$$

where n_{C_i} denotes the number of members in partitions C_i and $|C|$ is the number of partitions in C. If NMI equals one, the partition labels are equivalent to the ground-truth partition labels.

4.3. Results and comparative analysis

4.3.1. Experimental results of the proposed algorithm for determining c

In order to specify the number of communities (c), the proposed algorithm for determining c candidates with length K is executed on nine real-world networks, as well as the GN artificial network. Singular and REG values for the ten networks are shown in Figure 4, and REG_{RS} set of each network (as represented in Eq. (16)) is plotted versus REG values in Figure 5. According to Eq. (17) and Figure 5, K is equal to the index of maximum value of REG_{RS}. By considering the Karate network in Figure 5, the maximum value was observed for $K = 2$; therefore, there are two candidates for c. According to Eq. (14) and Figure 4, the set of c will be $\{2, 4\}$ for the Karate network. Similarly, there are $c \in \{2, 11, 12\}$ for Football network, $c \in \{2, 4\}$ for Jazz network, $c \in \{2, 3, 4\}$ for Dolphins network, $c \in \{2, 4, 6, 7, 9, 17\}$ for Citeseer network, $c \in \{2, 3\}$ for Polblogs network, $c \in \{2, 4, 6, 7\}$ for Cora network, $c \in \{2, 3, 4\}$ for Pubmed network, $c \in \{3\}$ for Polbooks network, and finally $c \in \{2, 4\}$ for GN network. By comparing the number of ground-true communities c in Table 1 with a set of c candidates of the ten networks, it is observed that for each network, $\bar{c}$ exists in the proposed set of candidates obtained from Eqs (14) and (17). For example, for the Karate network, $\bar{c} = 2$ and, therefore, $\bar{c} \in c = \{2, 4\}$. Consequently, the proposed algorithm for determining c candidates can find the right candidate q.

4.3.2. Performance comparison between Tpmtrinmf and Mtrinmf models

In this subsection, the efficiency improvement of the Tpmtrinmf model (by self-tuning β parameter) is compared to that of the Mtrinmf model in three parts. At first, it is shown that the proposed model is independent of the initial value of β in modularity index. Then, the performance improvement is illustrated based on the Tpmtrinmf model with respect to the Mtrinmf model. Afterward, it has been shown that the Tpmtrinmf algorithm could self-tune the β parameter.

One of the advantages of the Tpmtrinmf model is verified by the self-tuning β parameter. Therefore,

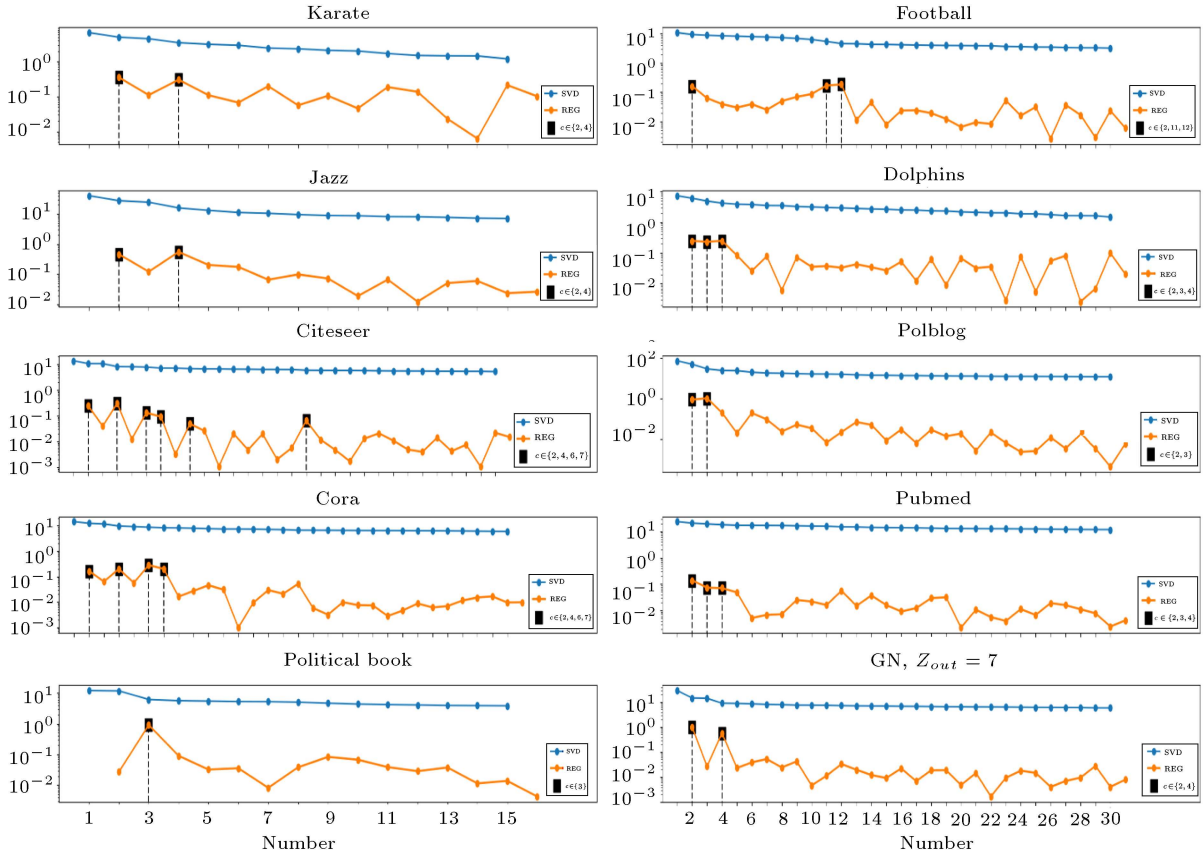

Figure 4. Singular and REG values of nine real networks and the GN artificial network. The set of c candidates is specified for each diagram on the network.

Figures 6 and 7 indicate the modularity information (Q) for different values of β and NMI for the two compared community detection models, respectively. Ten independent experiments with random initializations and the same inputs were executed for comparison. The nonlinear behaviors of the Mtrinmf model for different initial values of the β parameter for various datasets are shown in Figure 6. As seen, the behavior of the graphs shows three regimes for all types of networks: increasing, increasing with one extremum, and decreasing with one extremum. In other words, almost no constant graph is observed for Mtrinmf model. This observation is representative of the dependency of the Mtrinmf performance on the correct choice of the initial β value. On the other hand, it is clear from Figure 6 that the Tpmtrinmf model not only improves the modularity information average with respect to Mtrinmf model, but also is less sensitive to initial values of β. That is, the modularity average is almost independent of the initial values of β. Therefore, an arbitrary initial value for β can be selected with no worries about its convergence to the actual value.

According to Figure 7, since the Tpmtrinmf model uses only the modularity index, it cannot be independent of different initial values of β, which is not the case in the proposed Tpmtrinmf model.

In Figure 8, the time evolution of β for the two mentioned methods is indicated for the LFR artificial networks with $\mu = 0.3$ and seven real networks. According to Figure 8, the final values of β indicate that the proposed algorithm can powerfully handle various β initializations.

4.3.3. Performance analysis of Hamtrinmf method

In this section, the comparative results of the proposed Hamtrinmf method will be examined on real-world and artificial networks. The proposed method will be compared with other NMF-based methods, such as triNMF [32], Mtrinmf [25], $MHGNMF_{sq}$ [16], and DANMF [36]. The Mtrinmf model indicates that modularity information enhancement makes the Mtrinmf model perform better than the Trinmf model. So, the Mtrinmf method eliminates the requirement to set the candidate number of communities (c) by employing general Q_D. The $MHGNMF_{sq}$ model improves the NMF-based community detection by the structure similarity and topological connection information (hyperlaplacian matrix), and the DANMF model proposes a novel Deep Autoencoder approach to improve performance. Moreover, our method will be compared with other methods such as modularity-specialized Modularity-specialized label propagation

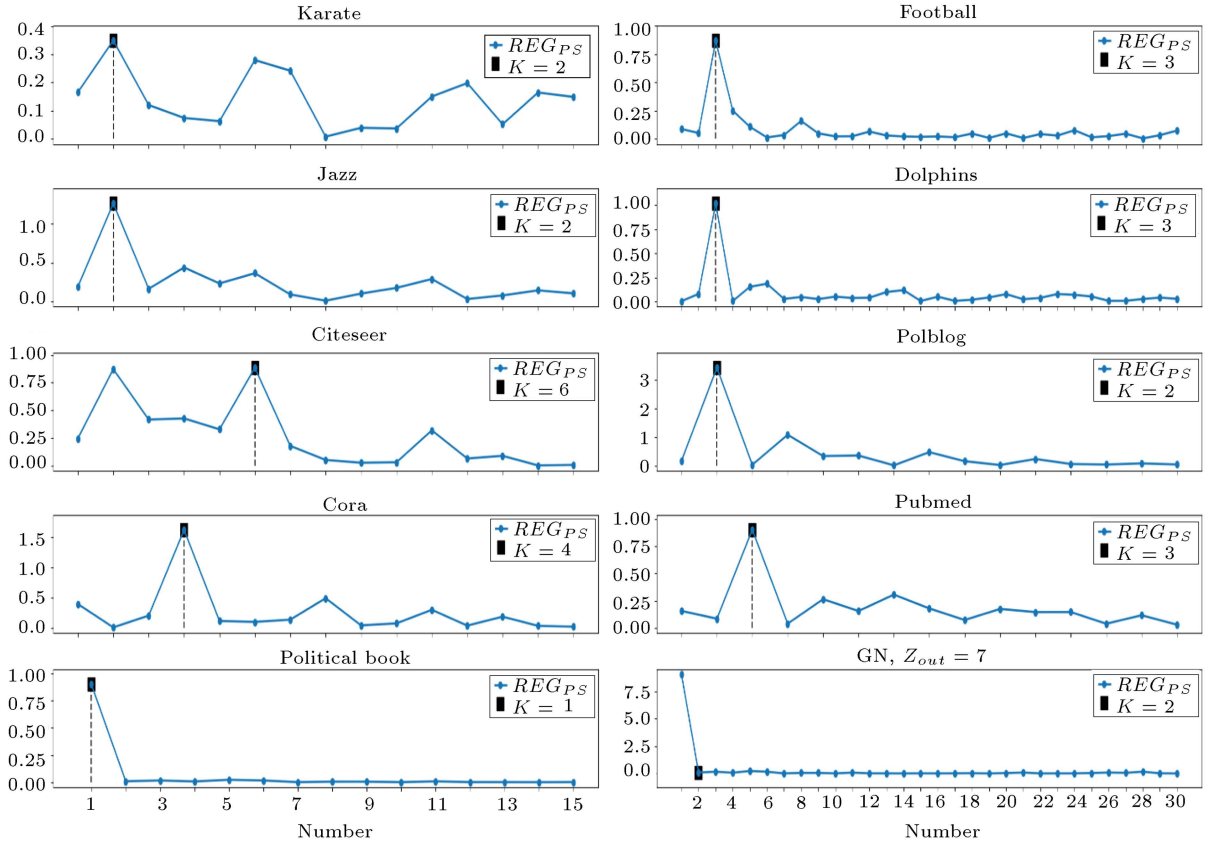

Figure 5. Identifying the values of K from REG_{RS} set for the ten networks.

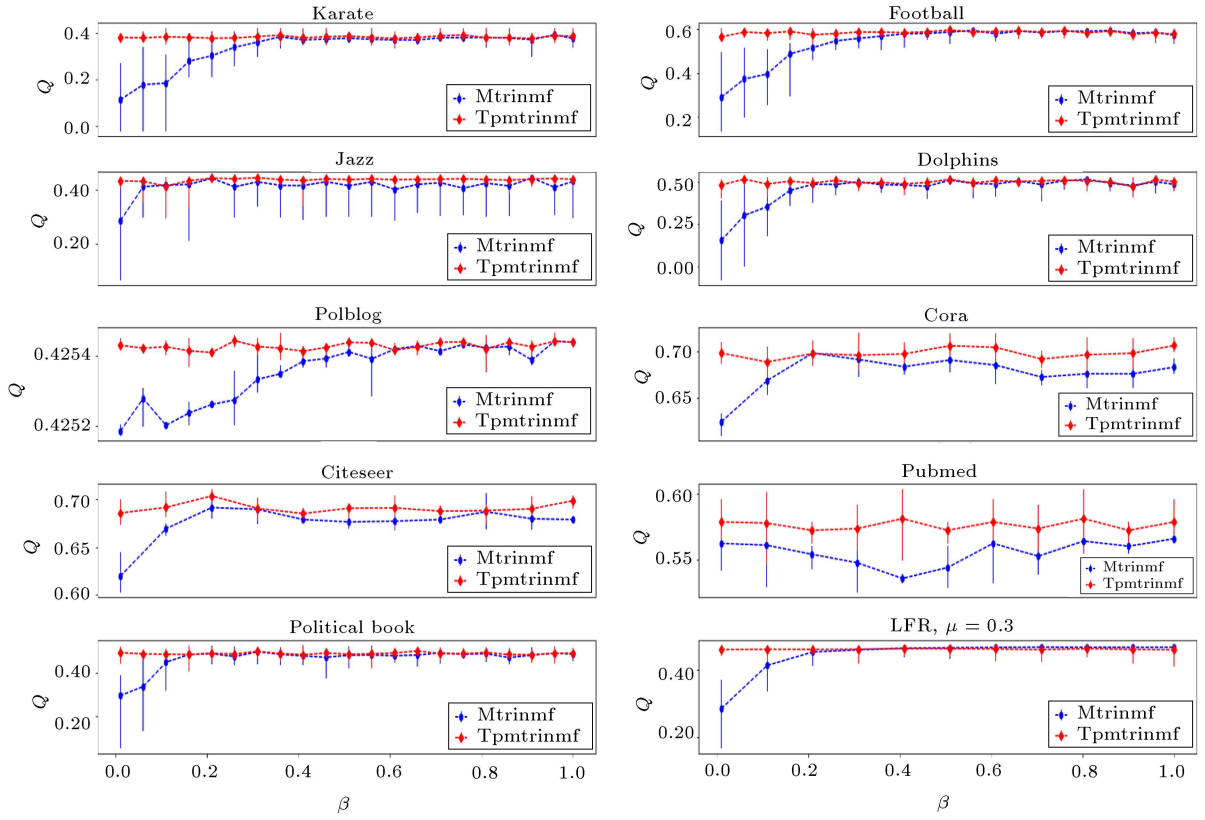

Figure 6. The modularity information (Q) of the two compared methods versus various initial values of β for different networks.

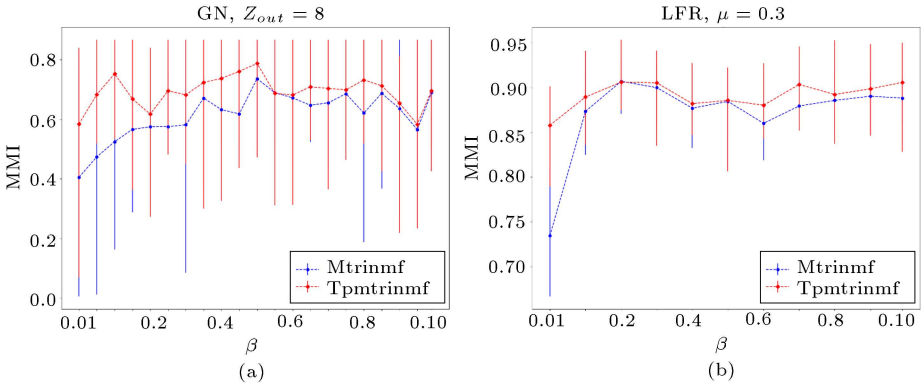

Figure 7. The NMI of two compared methods versus various initial values of β for GN and LFR networks.

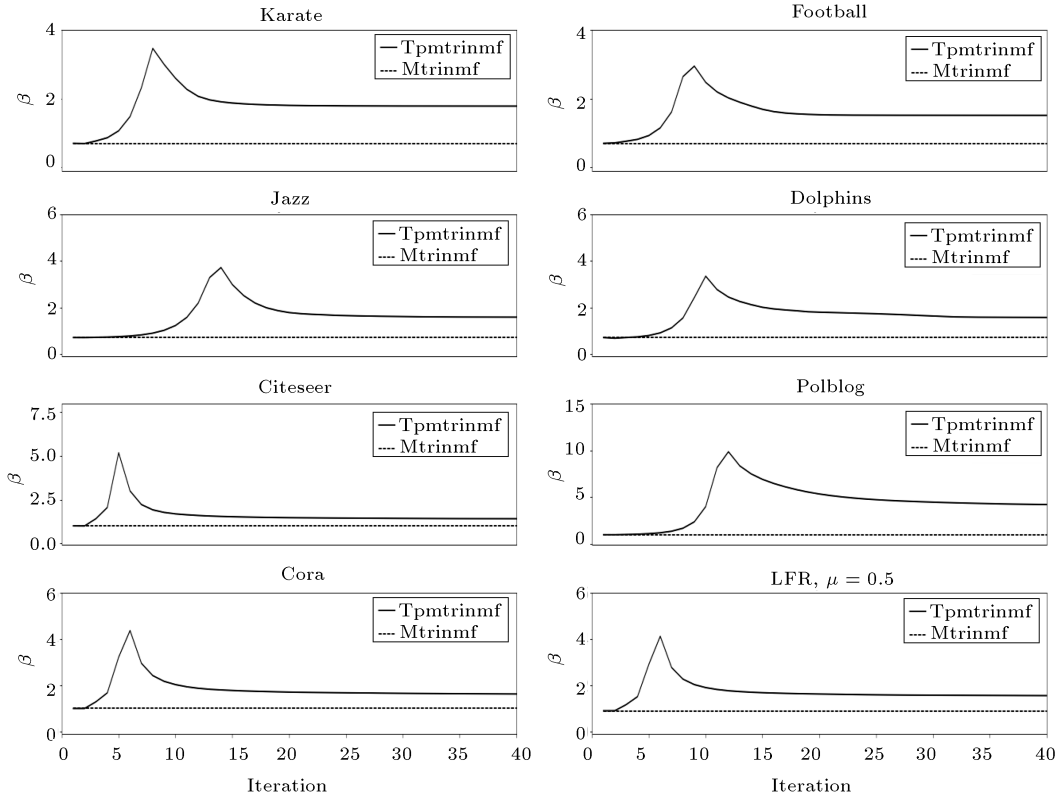

Figure 8. Time evolution of the self-tuned β of Tpmtrinmf compared to constant β of Mtrinmf models.

algorithm (LPAm) [45], CNM [46], Infomap [10], Louvain [47], and Low-rank Subspace based Network Community Detection (LRSCD) methods [48]. The LPA method obtains community detection based on node label propagation power [8,9]. Accordingly, Barber and Clark extended LPA and proposed the LPAm by relating it to modularity information [45]. The CNM is a popular community detection method based on fast greedy optimization to directly solve modularity information [46]. The Louvain method of community detection optimizes modularity locally on all nodes to find small communities and group each of them into one node [47]. Moreover, Infomap is another community detection method based on flow running dynamic by random walk [10], and LRSCD is a community detection model based on a low-rank decomposition strategy for decomposing each node vector in a new space (the geometric space) [48].

In Table 2, the best experimental results of compared methods are listed according to modularity information (Q) on nine real-world networks. In Figures 9 and 10, different c estimations and the comparative results based on NMI are presented for GN and LFR artificial networks by different methods. In Figure 11, ten experimental results on GN and LFR networks are shown for NMF-based community detection models. Note that since the Louvain and CNM methods perform similar community detection

Table 2. Comparison of modularity information (Q) for different methods and sets.

Method / Set	Hamtrinmf	Mtrinmf [25]	Trinmf [32]	$MHGNMF_{sq}$ [16]	LRSCD [48]	DANMF [36]	Infomap [10]	LPAm [45]	CNM [46]/ Louvain [47]
Karate	**0.419**	**0.419**	0.330	**0.419**	**0.419**	0.408	0.403	0.397	0.383
Jazz	**0.444**	0.442	0.423	**0.444**	0.442	0.441	0.442	**0.444**	**0.444**
Political books	**0.526**	**0.526**	0.481	**0.526**	0.520	0.512	**0.526**	0.520	0.508
Dolphins	**0.528**	0.526	0.338	0.526	0.526	0.511	0.520	0.518	0.498
Football	**0.605**	0.603	0.508	0.605	0.603	0.596	0.603	0.603	0.556
Polblogs	**0.42546**	0.42522	0.42446	0.4252	0.4252	0.4252	0.423	**0.4256**	**0.4256**
Cora	**0.701**	0.688	0.601	0.6663	0.629	0.647	0.231	0.526	0.660
Citeseer	0.766	0.712	0.630	0.712	0.630	0.687	**0.798**	0.551	0.724
Pubmed	0.594	0. 581	0.567	0.581	0.473	0.581	0.726	0.44	**0.751**

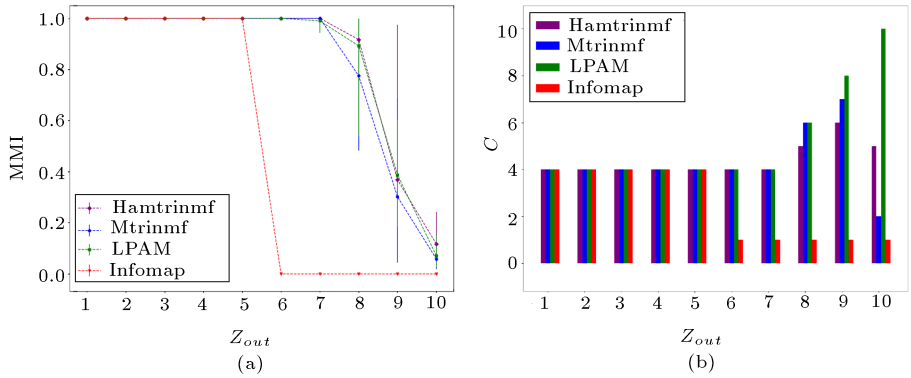

Figure 9. (a) Comparing the performance of different methods for GN networks based on NMI and (b) Number of communities (c) of GN network.

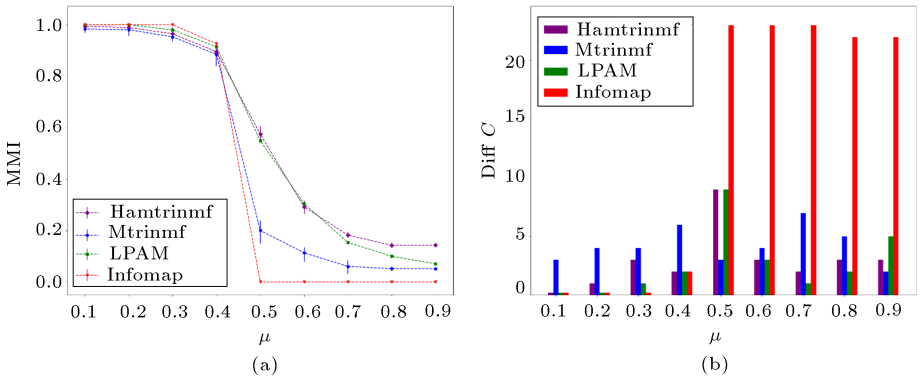

Figure 10. (a) Comparing the performance of different methods for LFR networks and (b) Difference between the number of communities (c) and ground-truth ($\bar{c} = 22$) ($Diffc = c - \bar{c}$).

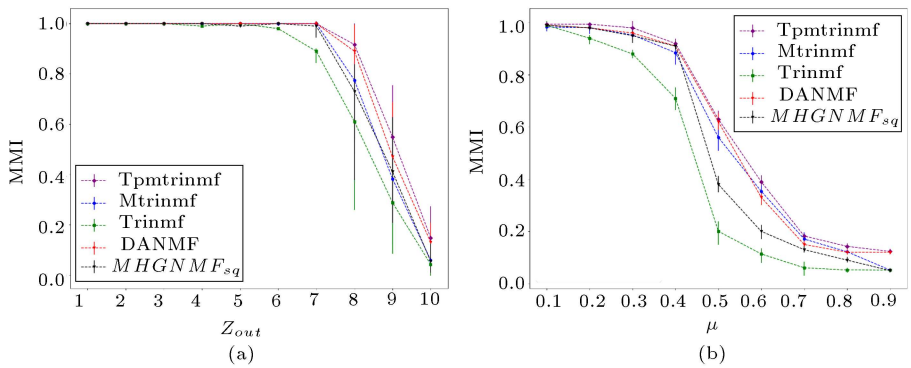

Figure 11. Performance comparison of different models for GN networks (a) and for LFR networks (b).

for all types of networks in Table 1, their results have been merged in one column in Table 2. However, the run times of these two methods are different and compared in the following. The results can be summarized as follows:

- By referring to Table 2, the Hamtrinmf method finds the highest modularity information (Q) and shows the best community detection based on NMF methods;
- In the Pubmed network in Table 2, CNM and Infomap have better clustering than other NMF-based methods. Due to the computational errors in large-scale networks, NMF-based community detection usually suffers from clustering errors. However, the Hamtrinmf performs better clustering than other methods;
- In Figure 9, the Hamtrinmf and LPAm methods outperform the Mtrinmf and Infomap approaches. Moreover, since c must converge to $\bar{c} = 4$ on GN network, the Hamtrinmf method has fewer detection errors versus different values of Z_{out} than other methods;
- In Figure 10, the Hamtrinmf method performs better than the Mtrinmf approach. At $\mu < 0.4$, the Infomap method outperforms other approaches. However, the Hamtrinmf and LPAm methods outperform the Mtrinmf and Infomap methods for $\mu > 0.5$;
- In Figure 11, by assuming the correct value of c, the Tpmtrinmf model improves the performance of the Mtrinmf model and provides better clustering than other NMF-based community detection models.

For comparison, the results of ten independent experiments with random initializations and the same inputs are shown in Figures 9 and 10.

4.4. Computational complexity analysis and comparison

In this section, Hamtrinmf and Mtrinmf methods are compared in terms of computational complexity. The computational complexity analysis of the Hamtrinmf method is conducted in three parts:

(i) Finding the complexity of determining a set of c candidates with length K;

(ii) Evaluating the complexity of executing the Tpmtrinmf model;

(iii) Computing the complexity of evaluation and selection of the best community.

In the first part, the main computational complexity includes two steps: (i) Estimating c candidates using SVD and REG algorithms and (ii) Selecting K largest values in the REG set. The complexity orders of SVD and REG algorithms are $\mathrm{O}\left(n^3\right)$ [27] and $\mathrm{O}\left(2h\right)$, respectively. In addition, the computational complexity is $\mathrm{O}\left(Kh\right)$. Since $h, K'' n$, the total complexity of this part will be $\mathrm{O}\left(n^3\right) + \mathrm{O}\left(2h\right). + \mathrm{O}\left(Kh\right) \approx \mathrm{O}\left(n^3\right)$. In the second part, the main computational complexity consists of calculating the updating rules H, W, and β. At one iteration, the complexity order of the updating rules H and W matrices is $\mathrm{O}\left(cn^2\right) + \mathrm{O}\left(c^2 n\right) + \mathrm{O}\left(c^2\right) + \mathrm{O}\left(cn\right)$ [25,48]. The updating rule for β parameter includes numerator, denominator, and conditions. By ignoring the calculated matrices such as $B_1 H$ and HWH^T, the complexity will be $\mathrm{O}\left(2nc\right) + \mathrm{O}\left(n^2\right) + \mathrm{O}\left(4nc\right)$ for all of them. Since $c'' n$, the total computational complexity of this part for I_t iteration is $\mathrm{O}\left(I_t cn^2\right) + \mathrm{O}\left(I_t c^2 n\right) + \mathrm{O}\left(I_t c^2\right) + \mathrm{O}\left(I_t cn\right) + \mathrm{O}\left(I_t n^2\right) \approx \mathrm{O}\left(I_t \left(c+1\right) n^2\right)$. In the third part, since the modularity density is used to select the best community, the complexity order of the community selection stage will be $\mathrm{O}\left(\left(c+3\right) n^2\right)$.

Finally, by applying the number of c candidates with length K, the total computational complexity of the Hamtrinmf method is $\mathrm{O}\left(KI_t \left(c+1\right) n^2\right) + \mathrm{O}\left(K \left(c+3\right) n^2\right) + \mathrm{O}\left(n^3\right)$. Although Hamtrinmf and Mtrinmf methods [25] are similar, they are different in choosing c candidates and selecting the best β parameter. By applying a different number of c candidates with length K' and K'' and different values of the β parameter, the total complexity of Mtrinmf method is $\mathrm{O}\left(K'' K' I_t cn^2\right) + \mathrm{O}\left(K'' K' \left(c+3\right) n^2\right)$. Karimi-Majd et al. [49] assumed $c \in \left\{2, 3, \ldots, \frac{\mathrm{n}}{3}\right\}$, whereas each cluster had more than three nodes. In addition, since $K' = \frac{n}{3}$ and $K'' K'$, the computational complexity of the Hamtrinmf method is smaller than the Mtrinmf method, implying that:

$$\begin{aligned} &\mathrm{O}\left(KI_t \left(c+1\right) n^2\right) + \mathrm{O}\left(K \left(c+3\right) n^2\right) \\ &\quad + \mathrm{O}\left(n^3\right) < \mathrm{O}\left(\frac{I_t cn^3}{3}\right) < \mathrm{O}\left(K'' K' I_t cn^2\right) \\ &\quad + \mathrm{O}\left(K'' K' \left(c+3\right) n^2\right). \end{aligned} \tag{30}$$

To compare the run times of different approaches, the run times of five methods and models on nine real-world and two synthetic networks were recorded, as shown in Tables 3 and 4. It is clear from Table 3 that the Louvain method is faster than other methods. Mtrinmf, on the other hand, has a longer execution time than others. Hence, from the run time perspective, our algorithm is inferior to CNM and Louvain, but better than Mtrinmf methods. The run times in Table 4 show that Tpmtrinmg is quite close to the $MHGNMF_{sq}$, Mtrinmf, and Trinmf models for a specific internal parameter, but is faster than DANMF. However, according to the results of Tables 1

Table 3. Comparison of run times (seconds) for different methods and sets.

Method / Set	Karate	Jazz	Political books	Dolphins	Football	Polblogs	Cora	Citeseer	Pubmed	GN	LFR
Hamtrinmf	0.871	4.22	34.86	1.36	2.69	51.74	1208.1	1680.0	61241	1.628	87.07
Mtrinmf [25]	11.670	65.61	981.96	7.58	14.96	454.30	5891.3	21098.7	–	7.205	483.29
Infomap [10]	0.203	0.91	0.86	0.28	0.97	4.13	14.7	16.6	4030	0.449	5.10
LPAm [45]	0.428	2.18	0.98	0.71	2.08	13.12	66.8	76.2	5837	0.891	24.84
CNM [46]	0.118	0.80	0.63	0.18	0.67	2.56	9.3	11.7	1916	0.127	3.12
Louvain [47]	0.062	0.47	0.56	0.92	0.29	1.09	5.9	4.9	1001	0.051	1.82

Table 4. Comparison of run times (seconds) for different models and sets.

Method / Set	Karate	Jazz	Political books	Dolphins	Football	Polblogs	Cora	Citeseer	Pubmed	GN	LFR
Tpmtrinmf	0.233	1.65	34.86	0.401	0.810	26.04	228.14	321.21	61241	0.361	23.17
$MHGNMF_{sq}$ [16]	0.225	1.28	27.16	0.368	0.705	21.12	189.10	245.32	28875	0.301	16.91
DANMF [36]	0.276	2.15	42.73	0.490	1.12	35.18	290.73	450.12	190890	0.440	34.99
Mtrinmf [25]	0.227	1.37	28.74	0.376	0.731	22.78	200.58	250.48	32510	0.313	20.07
Trinmf [32]	0.225	1.19	21.02	0.327	0.653	19.32	173.06	213.29	21653	0.290	15.15

to 4, one can conclude that our methods are more adaptable, more flexible, and less sensitive with better performance than other methods by using the tuning parameters approach and the algorithm of selecting the best number of communities, while it is fast enough, too.

The machine used for the present study is powered with Intel Core i7-6770 CPU and 8 GB RAM with 64-bit Windows 10 and Python (version 3.5) as the selected software.

5. Conclusions

This paper proposed a novel community detection method based on Non-negative matrix Factorization (NMF) called Hamtrinmf with an improved community detection performance and reduced computational complexity. First, the combination of Singular Value Decomposition (SVD) and the Relative Eigenvalue Gap (REG) was proposed to determine the set of candidates for the number of communities (c) with length K. Afterward, the Tuning parameter Mtrinmf (Tpmtrinmf) model was applied for community detection as an extension to Modularized tri-factor NMF (Mtrinmf) method using a self-tuning β parameter. Finally, general modularity density was used to obtain the best community in Hamtrinmf method. The experiment results of real-world and artificial networks demonstrated the efficiency of the proposed method. The computational complexity analysis proved that the Hamtrinmf method was faster than the Mtrinmf approach. In future works, the idea of self-tuning parameters can be generalized for other NMF-based community detection methods with the aim of both reducing their computational complexity and improving their performance. Moreover, it is possible to develop extended hybrid methods employing the combination of different similarity matrices of a graph, such as adjacency matrix, Laplacian matrix, etc., to improve the estimation of c.

References

1. Newman, M.E.J. "Networks", In *Oxford university Press* (2018).
2. Newman, M.E.J. and Girvan, M. "Finding and evaluating community structure in networks", *Phys. Rev. E.*, **69**, 026113 (2004).
3. Tang, W. and Daoutidis, P. "Network decomposition for distributed control through community detection in input-output bipartite graphs", *J. Process Control.*, **64**, pp. 7–14 (2018).
4. Campbell, W.M., Li, L., Dagli, C., et al. "Cross-domain entity resolution in social media", 4th Int. Workshop on Nat. Lang. Process. Social Media (2016).
5. Guerrero, M., Montoya, F.G., Baños, R., et al. "Adaptive community detection in complex networks using genetic algorithms", *Neurocomputing*, **266**, pp. 101–113 (2017).
6. Gong, M., MA, L., MA, Q., et al. "Community detection in networks by using multi-objective evolutionary algorithm with decomposition", *Phys. A: Stat. Mech. Appl.*, **391**(15), pp. 4050–4060 (2012).
7. Sun, P.G. "Community detection by fuzzy clustering", *Phys. A: Stat. Mech. Appl.*, **419**, pp. 408–416 (2015).
8. Cordasco, G. and Gargano, L. "Community detection via semi-synchronous label propagation algorithms", *Int. J. of Social Network Mining*, **1**(1), pp. 3–26 (2011).
9. Ding, J., He, X., Yuan, J., et al. "Community detection by propagating the label of center", *Phys. A: Stat. Mech. Appl.*, **503**, pp. 675–686 (2018).
10. Rosvall, M. and Bergstrom, C.T. "Maps of random walks on complex networks reveal community structure", *Proc. Natl. Acad. Sci. U.S.A.*, **105**(4), pp. 1118–1123 (2008).

11. Essid, S. and Fevotte, C. "Smooth nonnegative matrix factorization for unsupervised audiovisual documen structuring", *IEEE Trans. Multimedia*, **15**(2), pp. 415–425 (2013).

12. Peng, S., Ser, W., Chen, B., et al. "Semi-supervised nonnegative matrix factorization for image clustering", *Pattern Recognit*, **111**, 107683v (2021).

13. Huang, K., Fu, X., and Sidiropoulos, N.D. "Anchor-free correlated topic modeling: Identifiability and algorithm", *Adv. Neural Inf. Process. Syst.*, pp. 1794–1802 (2016).

14. He, C., Tang, Y., Liu, K., et al. "A robust multi-view clustering method for community detection combining link and content information", *Phys. A: Stat. Mech. Appl.*, **514**, pp. 396–411 (2018).

15. Wang, R.S., Zhang, S., Wang, Y., et al. "Clustering complex networks and biological networks by nonnegative matrix factorization with various similarity measures", *Neurocomputing*, **72**(1–3), pp. 134–141 (2008).

16. Wu, W., Kwong, S., Zhou, Y., et al. "Nonnegative matrix factorization with mixed hypergraph regularization for community detection", *Inf. Sci.*, **435**, pp. 263–281 (2018).

17. Binesh, N. and Rezghi, M. "Fuzzy clustering in community detection based on nonnegative matrix factorization with two novel evaluation criteria", *Appl. Soft Comput.*, **69**, pp. 689–703 (2018).

18. Yang, L., Cao, X., Jin, D., et al. "A unified semi-supervised community detection framework using latent space graph regularization", *IEEE Trans. Cybern.*, **45**(11), pp. 2585–2598 (2014).

19. Ma, X., Gao, L., Yong, X., et al. "Semi-supervised clustering algorithm for community structure detection in complex networks", *Phys. A: Stat. Mech. Appl.*, **389**(1), pp. 187–197 (2010).

20. He, C., Zhang, Q., Tang, Y., et al. "Community detection method based on robust semi-supervised nonnegative matrix factorization", *Phys. A: Stat. Mech. Appl.*, **523**, pp. 279–291 (2019).

21. Wang, F., Li, T., Wang, X., et al. "Community discovery using nonnegative matrix factorization", *Data Min. Knowl. Discov.*, **22**(3), pp. 493–521 (2011).

22. Zhang, Z.Y. "Nonnegative matrix factorization: Models, algorithms and applications", In *Data Mining: Foundations and Intelligent Paradigms. Intelligent Systems Reference Library*, Springer, Berlin, Heidelberg (2012).

23. Wang, Y.X. and Zhang, Y.J. "Nonnegative matrix factorization, A comprehensive review", *IEEE Trans. Knowl. Data Eng.*, **25**(6), pp. 1336–1353 (2013).

24. Liu, X., Wei, Y.M., Wang, J., et al. "Community detection enhancement using non-negative matrix factorization with graph regularization", Int. *J. Mod. Phys. B.*, **30**(20), 1650130 (2016).

25. Yan, C. and Chang, Z. "Modularized tri-factor nonnegative matrix factorization for community detection enhancement", *Phys. A: Stat. Mech. Appl.*, **533**, 122050 (2019).

26. Sarkar, S. and Dong, A. "Community detection in graphs using singular value decomposition", *Phys. Rev. E.*, **83**(4), 046114 (2011).

27. Lu, H., Sang, X., Zhoa, Q., et al. "Community detection algorithm based on nonnegative matrix factorization and pairwise constraints", *Phys. A: Stat. Mech. Appl.*, **522**, pp. 205–214 (2019).

28. Chen, K. and Lei, J. "Network cross-validation for determining the number of communities in network data", *J. Am. Stat. Assoc.*, **113**(521), pp. 241–251 (2018).

29. Wang, D. and Zhao, Y. "Network community from the perspective of time series", *Phys. A: Stat. Mech. Appl.*, **522**, pp. 205–214 (2019).

30. Fortunato, S. and Barthłemy, M. "Resolution limit in community detection", *Proc. Natl. Acad. Sci.*, U.S.A, **104**(1), pp. 36–41 (2007).

31. Huang, J., Zhang, T., Yu, W., et al. "Community detection based on modularized deep nonnegative matrix factorization", *Intern. J. Pattern Recognit. Artif. Intell.*, **35**(35), 2159006 (2021).

32. Wang, F., Li, T., Wang, X., et al. "Community discovery using nonnegative matrix factorization", *Data Min. Knowl. Discov.*, **22**(3), pp. 493–521 (2011).

33. Muff, S., Rao, F., and Caflisch, A. "Local modularity measure for network clusterizations", *Phys. Rev. E.*, **72**, 056107 (2005).

34. Li, Z., Zhang, S., Wang, R.S., et al. "Quantitative function for community detection", *Phys. Rev. E.*, **77**(3), 036109 (2008).

35. Zhang, M. and Zhou, Z. "Structural deep nonnegative matrix factorization for community detection", *Appl. Soft Comput.*, **97**, 106846 (2020).

36. Ye, F., Chen, C., Zheng, Z., et al. "Deep autoencoder-like nonnegative matrix factorization for community detection", *27th ACM Int. Conf.* (2018).

37. Luo, X., Liu, Z., Jin, L., et al. "Symmetric nonnegative matrix factorization-based community detection models and their convergence analysis", *IEEE Trans. Neural Netw. Learn. Syst.*, **99**, pp. 1–13 (2021).

38. Tasgin, M., Herdagdelen, A., and Bingol, H. "Community detection in complex networks using genetic algorithms", *arXiv*, 0711.0491 (2007).

39. Zachary, W.W. "An information flow model for conflict and fission in small groups", *J. Anthropol. Res.*, **33**(4), pp. 452–473 (1977).

40. Gleiser, P.M. and Danon, L. "Community structure in jazz", *Adv. Complex Syst.*, **6**(4), pp. 565–573 (2003).

41. Kunegis, J. "KONECT: The koblenz network collection", *Proc. 22nd Int. Conf. World Wide Web*, pp. 1343–1350 (2013).

42. Lusseau, D., Schneider, K., Boisseau, O.J., et al. "The bottlenose dolphin community of doubtful sound features a large proportion of long-lasting associations", *Behav. Ecol. Sociobiol.*, **54**(4), pp. 396–405 (2003).

43. Girvan, M. and Newman, M.E.J. "Community structure in social and biological networks", *Proc. Natl. Acad. Sci. U.S.A.*, **99**(12), pp. 7821–7826 (2002).

44. Lancichinetti, A., Fortunato, S., and Radicchi, F. "Benchmark graphs for testing community detection algorithms", *Phys. Rev. E.*, **78**(4) 046110 (2008).

45. Barber, M.J. and Clark, J.W. "Detecting network communities by propagating labels under constraints", *Phys. Rev. E.*, **80**, 026129 (2009).

46. Clauset, A., Newman, M.E.J., and Moore, C. "Finding community structure in very large networks", *Phys. Rev. E.*, **2**, 066111 (2004).

47. Blondel, V.D., Guillaume, J.L., Lambiotte, R., et al. "Fast unfolding of communities in large networks", *J. Stat. Mech. Theory Exp.*, **10**, pp. 1–12 (2008).

48. Ding, Z., Shang, Z., Sun, D., et al. "Low-rank subspace learning based network community detection", *Knowl. Based Syst.*, **155**, pp. 71–82 (2018).

49. Karimi-Majd, M., Fathian, A.M., and Amiri, B.A., "Hybrid artificial immune network for detecting communities in complex networks", *Comput.*, **97**(5), pp. 483–507 (2014).

Biographies

Mohammad Ghadirian was born in Iran in 1992. He received MSc degree in Electrical Engineering Majoring in Control from the Sharif University of Technology, Tehran, Iran in 2016. He is already a PhD candidate at the Electrical Engineering Department of Imam Khomeini International University, Qazvin, Iran. His research interest includes graph mining, data mining, and medical image processing.

Nooshin Bigdeli was born in 1978 in Iran and completed her PhD degree in Electrical Engineering majoring in Control at Sharif University of Technology, Tehran, Iran in 2007. She is currently an Associate Professor at the Electrical Engineering Department of Imam Khomeini International University, Qazvin, Iran. Her research interests include control systems, intelligent systems, chaos control, model predictive control, and model order reduction in high order systems.