Community detection using multilayer edge mixture model

Han Zhang¹ · Chang-Dong Wang¹ · Jian-Huang Lai¹ · Philip S. Yu²

Received: 1 November 2016 / Revised: 18 March 2018 / Accepted: 10 May 2018 / Published online: 8 June 2018
© Springer-Verlag London Ltd., part of Springer Nature 2018

Abstract Multilayer networks are networks where edges exist in multiple layers that encode different types or sources of interactions. As one of the most important problems in network science, discovering the underlying community structure in multilayer networks has received an increasing amount of attention in recent years. One of the challenging issues is to develop effective community structure quality functions for characterizing the structural or functional properties of the expected community structure. Although several quality functions have been developed for evaluating the detected community structure, little has been explored about how to explicitly bring our knowledge of the desired community structure into such quality functions, in particular for the multilayer networks. To address this issue, we propose the multilayer edge mixture model (MEMM), which is positioned as a general framework that enables us to design a quality function that reflects our knowledge about the desired community structure. The proposed model is based on a mixture of the edges, and the weights reflect their role in the detection process. By decomposing a community structure quality function into the form of MEMM, it becomes clear which type of community structure will be discovered by such quality function. Similarly, after such decomposition we can also modify the weights of the edges to find the desired community structure. In this paper, we apply the quality functions modified with the knowledge of MEMM to different multilayer benchmark networks as well as real-world multilayer networks and the detection results confirm the feasibility of MEMM.

Keywords Community detection · Multilayer network · General quality function · Edge mixture

Chang-Dong Wang
changdongwang@hotmail.com

¹ School of Data and Computer Science, Sun Yat-sen University, Guangzhou, People’s Republic of China
² University of Illinois at Chicago, Chicago, IL 60607, USA
1 Introduction

Networks have been widely used in characterizing complex systems in various areas such as transportation networks, electrical networks, social networks, and biological networks [16, 21, 26, 32, 43, 46–48]. Traditionally, a network is represented as a graph where the nodes represent individuals of the network and the presence of an edge between a pair of nodes indicates their connection [5]. In a more complex scenario, a variety of attributes of the edges are explored, which lead to directed graphs [1, 32], weighted graphs [2, 33], signed graphs [13, 49] etc. Although these graphs have successfully depicted a wide range of network systems, they fail to comprehensively construe the network structure when the edges are distinguished into multiple types or the network is temporal (i.e., the edges of the network vary over time) [19, 40, 44, 45]. For instance, consider a phone call network where the nodes represent the users and the edge between them indicates a phone call. Obviously, the links in the system evolves over time. Then if there is a link observed between two individuals in successive time slices, they may have a long phone call that lasts for a while or several independent calls. To eliminate such ambiguity, the interdependency of the time slices should be taken into account. In recent years, such networks with multiple interdependent “layers” which are represented by a series of graphs that describe the network from different perspectives have sprung rapidly especially in transportation, gene and online networks [6, 27, 44]. During the process of exploring the multilayer networks, different network representations have been explored [24]. In this paper, we will consider the “multiplex network” model for representation of multilayer networks, which is a typical representation referenced in related researches [4, 15, 31], as shown in Fig. 1. This model assumes that the layers share the same node set and are linked by the couplings between a node in one layer and its counterparts in other layers. Recall the example of phone call network. The ambiguity issue arisen from the continuity of the calls is well addressed by introducing couplings between the users having long phone calls.

Fig. 1 Illustration of the representative multilayer network model, a.k.a. the multiplex network. All the layers share the same node set and are represented by a simple graph. The layers are coupled by the “couplings” between one node and its copy in other layers, plotted as the dotted lines. It is also assumed that there is no edges connecting different physical nodes (i.e., different nodes in the common node sets shared by all layers) in different layers.
A major interest of complex network study is focused on discovering the community structure, which helps interpret the structure of network data, especially those with large size [30]. A community is a group of nodes that interact in a stochastically equivalent way with other groups of nodes, albeit there is actually no universal definition of a “community”, since the concept of a community is application-dependent [14]. For instances, there are two types of community structure, namely assortative community structure [34] and disassortative community structure [29]. In the assortative community structure, a community usually refers to a group of nodes that are densely connected with each other and sparsely connected with the nodes outside the group. Therefore, based on different assumption on the characterizations of the communities, numerous quality functions for community structure have been developed to evaluate the detected community structure [34,35,37–39,41,42,50]. Many existing community detection methods choose a specific quality function and optimize it with respect to the community assignments for the nodes. The community assignments corresponding to the optimal quality value are recognized as the optimal partition.

In consideration of numerous existing objective functions and the subtleties of the community definition, it is necessary to understand the differences and advantages of the existing methods. One way to achieve this goal is to represent the existing quality functions in an explicable form before comparison. In this paper, inspired by the rewarding scheme proposed by Reichardt and Bornholdt [38], we propose the multilayer edge mixture model (MEMM) which is positioned as a general framework that enables us to design a quality function that reflects our knowledge about the desired community structure. The contributions of this model are twofold:

1. The framework enables straightforward design of new quality functions given prior knowledge of the very definition of “community” in a real-world application.
2. The framework deepens our understanding of the existing quality functions for community detection, based on which the comparison between quality functions and the extension to the original method becomes easier.

In this paper, we will demonstrate the use of MEMM by taking the multilayer modularity as an example, which is one of the most popular methods to find communities in multilayer networks. By representing the multilayer modularity in the form of MEMM, we give some new interpretations to it as well as some possible modifications. As confirmed by experiments, the main findings can be concluded as follows.

- The current multilayer modularity has trouble detecting communities in networks containing disassortative communities, which have loose intra-community edges and dense inter-community edges. After modifying the multilayer modularity according to MEMM, the quality function is capable of detecting such communities.
- In addition, when the community structure varies among layers, the robustness of modularity can be improved by modifying the coupling contribution.
- Finally, experiments on real-world networks confirm that the parameters introduced by MEMM are able to adjust the detection preference, including the contribution balance between edges and couplings, as well as the community consistency and detection granularity among layers.

The rest of this paper will be arranged as follows. We briefly review the related work in the literature in Sect. 2. We will introduce the multilayer edge mixing model in Sect. 3. The experimental results are reported in Sect. 4. We conclude this paper in Sect. 5.
2 Related work

To the best of our knowledge, the major quality functions applied to the multilayer community detection problem include modularity-based methods [3, 31], stochastic blockmodels (SBMs) [36] and information-theoretic methods [10].

Modularity evaluates the quality of a network partition, where a higher modularity value usually indicates a denser edge distribution within communities [7, 32, 34, 35]. Furthermore, Delvenne et al. derived the multilayer modularity by assessing the capability of the given community structure to capture a dynamic process in a multilayer network [11]. They compare the probability of a random walker to stay within a community after a series of jumps with the steady state of the normalized Laplacian dynamic (i.e., the probability of staying in the community after infinite jumps) and find that the first-order approximation of the matrix exponential in this difference leads to a similar form of the modularity in single-layer networks. Therefore, the communities that keep trapping the random walker will achieve a relatively high modularity score.

In the literature, a stochastic blockmodel (SBM) refers to the statistical models where the edges are considered as random variables sampled from specific probability distributions [18, 23]. SBMs assume that edges connecting the nodes of the same community are sampled from the same distribution. Edges lying between communities are also sampled from the same distribution. Therefore, by modeling such probability distributions with community labels as latent variables, we are able to infer the community labels for all the nodes [23] and detect missing edges [17], by using the observed linking structure as evidence. The probability distribution adopted to generate the edges lies at the heart of SBMs. Improper choice of the probability distribution can lead to rather poor detection results. Since the discussion about the specific choice of probability distribution is beyond the scope of this paper, we point the readers to [23].

Information-theoretic methods take another perspective on the issue of community structure. In real world, when we assign names to every road in the country, instead of assigning each road a unique name, we usually reuse common names such as “Park Street” in different cities. Similarly, when we are trying to encode the structure of a network or a dynamic process on it, reusing codenames within communities can greatly save coding cost for transmission [41, 42]. De Domenico et al. proposed a multilayer infomap method which generalizes this idea to the multilayer case [10]. The idea is transforming the community detection problem to a dual-coding problem by utilizing the community structure to best compress the bytes needed to characterize a random walk process on the network.

A wide variety of quality functions have been proposed to solve the community detection problem from different perspectives. Unfortunately, little has been brought to light about the similarities and differences of these quality functions, i.e., what is the difference between the community structures detected with respect to these quality functions? To provide insight into this question, in this work, we propose a multilayer edge mixture model (MEMM) to explore a relatively general representation of the community structure quality functions for the multilayer community structure from an edge mixture view. In particular, we will focus on translating the widely used multilayer modularity measure to our framework. We will see in the later sections that by representing a specific quality function as the form of MEMM, we are able to tell what kind of community structure will be discovered by the quality function so the comparison to other functions or possible extensions becomes possible.
3 The multilayer edge mixture model

In [38], Reichardt and Bornholdt proposed a rewarding scheme of edges to describe a general quality function for community structure in single-layer networks: (i) rewarding existing edges within a community, (ii) penalizing non-existing edges within a community, (iii) penalizing existing edges between two communities and (iv) rewarding non-existing edges between two communities. Given the edge strength matrix \( A = [A_{ij}] \in \mathbb{R}^{N \times N} \) of a single-layer network consisting of \( N \) nodes with \( A_{ij} \) being the edge strength of nodes \( i \) and \( j \), the general quality function of the rewarding scheme is as follows:

\[
\mathcal{H}(\mathbf{v}) = -\sum_{i \neq j} a_{ij} A_{ij} \delta(v_i, v_j) + \sum_{i \neq j} b_{ij} (1 - A_{ij}) \delta(v_i, v_j) + \sum_{i \neq j} c_{ij} A_{ij} [1 - \delta(v_i, v_j)] - \sum_{i \neq j} d_{ij} (1 - A_{ij}) [1 - \delta(v_i, v_j)].
\]

(1)

In the above equation, the notations are defined as follows:

- The vector \( \mathbf{v} = [v_i] \in \mathbb{R}^N \) is a community assignment vector of nodes with \( v_i \) being the community label of node \( i \).
- The matrix \( a = [a_{ij}] \in \mathbb{R}^{N \times N} \) (resp., \( b = [b_{ij}] \in \mathbb{R}^{N \times N} \), \( c = [c_{ij}] \in \mathbb{R}^{N \times N} \), \( d = [d_{ij}] \in \mathbb{R}^{N \times N} \)) is a parameter matrix with \( a_{ij} \) (resp., \( b_{ij} \), \( c_{ij} \), \( d_{ij} \)) controlling the rewarding/penalizing strength of internal existing edges (resp., internal non-existing edges, external existing edges, external non-existing edges).
- The delta function \( \delta \) is the Kronecker delta, i.e., \( \delta(x, y) = 1 \) if \( x = y \), otherwise \( \delta(x, y) = 0 \).

A lower \( \mathcal{H}(\mathbf{v}) \) value indicates a better partition. Despite great success, such rewarding scheme is improper for some networks, e.g., disassortative networks, where edges are expected to distribute between the communities rather than within them. In fact, in such networks the absent edges should be encouraged instead. This implies that using the signs to assert the contribution type is too rigid to extend to other network types. In addition, the parameter matrixes \( a, b, c, d \) are restricted to be positive, which limits the form they can take. Another quality function, the role model, can be rewritten in the same form [39].

Rather than directly finding the community assignment for each node, we first introduce parameters and determine their values according to the definition of the “community”, to obtain a quality function of the community structure. Then the quality function evaluates how significant the detected community structure is and guides us to the optimal community assignment. In this paper, we call a model taking such a two-phase approach to determine the community assignment a hyper model to distinguish them from conventional models in the literature, which usually refer to specific quality functions. Such hyper models act as a framework for “generating” a conventional model.

3.1 The model

Inspired by the previous work, in multilayer networks, we propose the multilayer edge mixture model (MEMM), which is a multilayer hyper model that employs couplings connecting pairs of layers and introduces the probabilities on the links. Given a multilayer network consisting of \( N \) nodes and \( l \) layers, the intra-layer edge strength matrix and the inter-layer coupling strength matrix are denoted as \( A = [A_{ijs}] \in \mathbb{R}^{N \times N \times l} \) and \( C = [C_{ijsr}] \in \mathbb{R}^{N \times l \times l} \), respectively, with \( A_{ijs} \) denoting the intra-layer edge strength of nodes \( i \) and \( j \) in layer \( s \) and
C_{isr} denoting the inter-layer coupling strength of node $i$ between layers $s$ and $r$. The general quality function of MEMM is as follows:

$$
\mathcal{M}(\nu) = \sum_{i \neq j, s} \lambda_a(a_{isj})A_{isj}P(v_{is}, v_{js}) + \sum_{i \neq j, s} \lambda_h(b_{isj})(1 - A_{isj})P(v_{is}, v_{js}) \\
+ \sum_{i \neq j, s} \lambda_c(c_{isj})A_{isj}[1 - P(v_{is}, v_{js})] + \sum_{i \neq j, s} \lambda_d(d_{isj})(1 - A_{isj})[1 - P(v_{is}, v_{js})] \\
+ \sum_{s \neq r, i} \lambda_e(e_{isr})C_{isr}P(v_{is}, v_{ir}) + \sum_{s \neq r, i} \lambda_f(f_{isr})(1 - C_{isr})P(v_{is}, v_{ir}) \\
+ \sum_{s \neq r, i} \lambda_g(g_{isr})C_{isr}[1 - P(v_{is}, v_{ir})] + \sum_{s \neq r, i} \lambda_h(h_{isr})(1 - C_{isr})[1 - P(v_{is}, v_{ir})].
$$

(2)

In the above equation, the notations are defined as follows:

- The indexes $i, j$ (resp., $s, r$) indicate nodes (resp., layers), and node $is$ means node $i$ in layer $s$.
- The matrix $\nu = [v_{is}] \in \mathbb{R}^{N \times l}$ is a community assignment matrix of multilayer nodes with $v_{is}$ being the community label of node $i$ in layer $s$.
- The matrix $\mathbf{a} = [a_{isj}] \in \mathbb{R}^{N \times N \times l}$ is an intra-layer parameter matrix with $a_{isj}$ controlling the contribution of the corresponding internal existing edge. Similarly for the remaining three intra-layer parameter matrices of size $N \times N \times l$, i.e., $\mathbf{b} = [b_{isj}]$, $\mathbf{c} = [c_{isj}]$, $\mathbf{d} = [d_{isj}]$.
- The matrix $\mathbf{e} = [e_{isr}] \in \mathbb{R}^{N \times l \times l}$ is an inter-layer parameter matrix with $e_{isr}$ controlling the contribution of the corresponding internal existing coupling. Similarly for the remaining three inter-layer parameter matrices of size $N \times l \times l$, i.e., $\mathbf{f} = [f_{isr}]$, $\mathbf{g} = [g_{isr}]$ and $\mathbf{h} = [h_{isr}]$.
- The probability matrix $\mathbf{P} = [P(v_{is}, v_{jr})] \in \mathbb{R}^{N \times l \times N \times l}$ with $P(v_{is}, v_{jr}) \equiv \Pr(v_{is} = v_{jr})$ being the probability of how evident node $is$ and node $jr$ belong to the same community.
- The $\lambda_a(a_{isj})$ function indicates whether we encourage or discourage the corresponding internal existing edge with the parameter $a_{isj}$:

$$
\lambda_a(a_{isj}) = \begin{cases} 
+ a_{isj} & \text{if we encourage such edge;} \\
- a_{isj} & \text{otherwise}.
\end{cases}
$$

(3)

Similarly for the remaining seven indicator functions, i.e., $\lambda_h(b_{isj})$, $\lambda_c(c_{isj})$, $\lambda_d(d_{isj})$, $\lambda_e(e_{isr})$, $\lambda_f(f_{isr})$, $\lambda_g(g_{isr})$, $\lambda_h(h_{isr})$.

Unlike the treatment in Eq. (1), we introduce the $\lambda$ functions so that the parameters can take values with arbitrary signs. Meanwhile, using the $\lambda$ functions rather than the signs of the terms makes MEMM more flexible to deal with different network structure.

By using a probability representation $P(v_{is}, v_{jr})$ rather than the $\delta$ function like Eq. (1), MEMM enables a “fuzzy” partition of the network in some cases, as we do not assert that two nodes belong to either the same or different communities [51]. We can easily revert to a “hard” division by rounding the entries of the probability $\mathbf{P}$ to binary values. Notice that the mixture coefficients are not required to be independent from the probability.

© Springer
Note that MEMM is not based on the exact definition of the community or the probability distribution. It just encourages the presence (or absence) of the existing (or non-existing) of the edges (or couplings) within (or between) communities, weighted by the parameters and the corresponding linking probabilities. Moreover, MEMM formally describes the quality of community structure as a mixture of eight types of edges. By representing a quality function in this way, the community type this quality function tends to discover can be reflected by the corresponding weights, e.g., whether the cohesion or adhesion of the community is preferred.

Before studying the community assignments, we need to determine eight parameter matrices \( \{a, b, c, d, e, f, g, h\} \), which control the contribution of the corresponding edges (couplings). Two strategies can be used to choose the parameters, i.e., to take them as fixed values or specific functions of the network structure. We may assign fixed values to the parameters once we have determined the contribution proportion of the edges (couplings) in the network. For example, by taking \( \lambda c(c_{ijs}) = -1 \) for all \( i, j, s \), \( P(v_{is}, v_{js}) = \delta(v_{is}, v_{js}) \) and omitting other terms in Eq. (2), we obtain the cut size. Another choice is to take the parameters as a function of the topology (i.e., \( A, C \)) or even the community structure (i.e., \( P \)). It is worth noticing that if the parameters have different domains (for example, the parameter matrix \( a \) can take a sufficiently larger magnitude than other parameters), the model may not function as expected, which just consists of the preferred terms, since even the largest values other parameters can take will be overpowered by the terms with the parameter matrix \( a \). As a result, we should choose the parameters for MEMM in a manner that guarantees distinguishing the contribution of different edges and couplings while avoiding being dominated by specific terms. In MEMM, this requires the choice of the parameters to be discriminative: i) if we choose two parameters to take the largest value they can take, equal attention is paid to both terms; ii) if we need to emphasize a specific edge type from the others, we just make that term take a relatively large value in its domain and others take a relatively small value in their own domains. Note that the terms with smaller weights still have the “potential” to take effect. Accordingly, we have the definition on the discriminative condition as follows.

**Definition 1** A parameter matrix set \( S = \{a, b, c, d, e, f, g, h\} \) of MEMM is discriminative if for any two matrices \( \omega_1, \omega_2 \in S \),

\[
\frac{\sup_{\text{dom}(\omega_1)} |\forall t_1 \in \omega_1|}{\sup_{\text{dom}(\omega_2)} |\forall t_2 \in \omega_2|} = 1.
\]

where \( \text{dom}(\omega_1) \) (resp., \( \text{dom}(\omega_2) \)) denotes the value range of all entries in matrix \( \omega_1 \) (resp., \( \omega_2 \)), and \( \sup_{\text{dom}(\omega_1)} |\forall t_1 \in \omega_1| \) (resp., \( \sup_{\text{dom}(\omega_2)} |\forall t_2 \in \omega_2| \)) denotes the largest absolute value that any entry in matrix \( \omega_1 \) (resp., \( \omega_2 \)) can take. Here we assume \( \infty/\infty = 1 \) to make the notations uncluttered.

Definition 1 states that the edges (couplings) should have exactly the same contribution when the absolute value of the corresponding parameters approximate their supremum (the same infinity norm). This means by selecting a discriminative parameter matrix set, the terms of MEMM have equal potential to affect the global objective function value. Discriminative parameter matrix sets guarantee the parameters are not dominating each other.

**3.2 Deriving modularity from MEMM**

Muchait et al. proposed the multilayer modularity based on a Laplacian dynamic defined on the multilayer networks [31]. They assume that a random walker tends to stay in the same community after repeated random jumps. Here we demonstrate that MEMM provides another interpretation of the multilayer modularity.
If we take \( \lambda_a(a_{ijs}) = +a_{ijs}, \lambda_b(b_{ijs}) = -b_{ijs}, \lambda_c(c_{ijs}) = -c_{ijs}, \lambda_d(d_{ijs}) = +d_{ijs}, \lambda_e(e_{isr}) = +e_{isr}, \lambda_f(f_{isr}) = -f_{isr}, \lambda_g(g_{isr}) = -g_{isr}, \lambda_h(h_{isr}) = +h_{isr}, \) and

\[
\begin{align*}
    a_{ijs} &= c_{ijs} = 1 - \gamma_s p_{ijs} \\
    b_{ijs} &= d_{ijs} = \gamma_s p_{ijs} \\
    e_{isr} &= g_{isr} = \varsigma \\
    f_{isr} &= h_{isr} = 0,
\end{align*}
\]

where \( p_{ijs} \) is the null model in layer \( s \), i.e., a random graph after rewiring the edges in the original graph. The \( \gamma_s \) is the resolution parameter in layer \( s \) and \( \varsigma \) controls the coupling strength between the layers. Ignoring the contribution of non-existing couplings here, we obtain

\[
\mathcal{M}(\nu) = \sum_{i \neq j,s} (A_{ijs} - \gamma_s p_{ijs})[2P(\nu_{is}, \nu_{js}) - 1] + \sum_{s \neq i,r} \varsigma C_{isr}[2P(\nu_{is}, \nu_{ir}) - 1]
\]

\[
= 2 \sum_{isr} \left[(A_{ijs} - \gamma_s p_{ijs})P(\nu_{is}, \nu_{js}) + \tilde{C}_{isr} P(\nu_{is}, \nu_{ir})\right] - \sum_{s}(1 - \gamma_s)m_s - \sum_{i} m'_i,
\]

where we utilize the fact that \( \sum_{ij} A_{ijs} = \sum_{ij} p_{ijs} \) and define \( \tilde{C}_{isr} = \varsigma C_{isr}, m_s = \sum_{ij} A_{ijs} \) and \( m'_i = \sum_{sr} \tilde{C}_{isr} \) to keep the notations uncluttered. Although beyond the scope of this paper, a “fuzzy” modularity representation can be obtained from MEMM as a by-product, where the nodes are considered to be in the same community with a probability, and this reflects how reliable the current assignment is. In this way, the summation of effective edges now runs throughout the network rather than merely within the communities, and the optimization requires exploring the possible settings of \( P \) where all entries take continuous value in \([0, 1]\). If we take a hard partition, maximizing \( \mathcal{M}(\nu) \) is then equivalent to the optimization of the multilayer modularity proposed by Muchait et al.

\[
\mathcal{M}(\nu) = 2 \sum_{isr} \left[(A_{ijs} - \gamma_s p_{ijs})\delta_{sr} + \tilde{C}_{isr}\delta_{ij}\right] \delta(\nu_{is}, \nu_{jr}) - \sum_{s}(1 - \gamma_s)m_s - \sum_{i} m'_i,
\]

where the \( \delta \) function is the Kronecker delta.

Now let’s take a closer look at the derivation. The contribution of the intra-layer edges is controlled by the resolution limit \( \gamma_s \) with range \([\pm(1 - \gamma_s p_{ijs}), \pm \gamma_s p_{ijs}]\). If we assume \( \gamma_s \in [0, 1] \), the weight range of the extreme value within each layer is \([\pm 1, \pm \gamma_s] \). Therefore, the discriminative condition in Definition 1 is satisfied only when \( \gamma_s = \varsigma = 1 \). Using other parameter settings for \( \gamma_s \) and \( \varsigma \) implicitly adjusts the potential for external existing edges to take effect. The extremal circumstance is taking \( \gamma_s = 0 \), which is equivalent to completely omit the punishment of the external existing edges, and \( \gamma_s = \infty \), which is equivalent to completely forbid the existence of internal edges. Recall from Eq. (4) that the probability of linking in the null model of each layer is represented as \( p_{ijs} \), weighted with the resolution limit \( \gamma_s \). The difference \( A_{ijs} - \gamma_s p_{ijs} \) can be interpreted as the expected
edge strength between node $is$ and $js$ with respect to a randomly rewired network with $p_s$ as adjacency matrix. The effective edge strength of the observed network takes the value $A_{ijs} - \gamma_s p_{ijs} \in \{ -\gamma_s p_{ijs}, 1 - \gamma_s p_{ijs} \}$. We find that the effective edge strength matches the value of the parameters [in Eq. (4)] we pick to derive the modularity. This suggests that the parameters here act as reconstructing the network adjacency matrix of the observed network. Different choice of the null model $p_{ijs}$ will lead to different preference on the community structure. For example, if we choose the Newman–Girvan null model $k_{is} k_{js} / 2m_s$ [34], where $k_{is}$ denotes the degree of node $is$ within the layer and $2m_s = \sum_i k_{is}$, the contribution of the edges between nodes with high degrees will get punished more heavily. The edges between nodes with large degree difference or nodes with low degrees will contain more information of the community structure.

It is worth noticing that the choice of the parameters is not unique. Indeed, if we take $\lambda_d(a_{ijs}) = +a_{ijs}, \lambda_e(b_{ijs}) = -b_{ijs}, \lambda_e(e_{isr}) = +e_{isr}$, with

$$
\begin{align*}
    a_{ijs} &= 1 - \gamma_s p_{ijs} \\
    b_{ijs} &= \gamma_s p_{ijs} \\
    e_{isr} &= \varsigma \\
    c_{ijs} &= d_{ijs} = f_{isr} = h_{isr} = g_{isr} = 0,
\end{align*}
$$

or take $\lambda_d(c_{ijs}) = -c_{ijs}, \lambda_d(d_{ijs}) = +d_{ijs}, \lambda_e(e_{isr}) = +e_{isr}$, with

$$
\begin{align*}
    c_{ijs} &= 1 - \gamma_s p_{ijs} \\
    d_{ijs} &= \gamma_s p_{ijs} \\
    e_{isr} &= \varsigma \\
    a_{ijs} = b_{ijs} = f_{isr} = h_{isr} = g_{isr} = 0,
\end{align*}
$$

we obtain the same optimization objective function as in Eq. (5) except the constant terms. This indicates that the modularity can be represented subject to either the internal or external edges within the layers, as explored in the literature [12,34]. But in either way, we know that the modularity favors the community structure with dense within-community edges and sparse between-community edges.

We have derived the multilayer modularity from MEMM by ignoring the absent couplings. Using MEMM, however, we can also take non-existing couplings into account by setting $f_{isr} = h_{isr} = \varsigma$. This will lead to a different coupling contribution, where $\tilde{C}_{isr} = \varsigma (2C_{isr} - 1)$. Then $\tilde{C}_{isr}$ can take negative values, which decreases the global modularity value. We will examine the difference of these two weighting schemes for the couplings in experiments. Rather than utilizing a dynamic process, we naturally obtain the multilayer modularity based on MEMM by comparing the edge distribution of the observed network with the null model, which returns to the original definition of modularity [34]. Similar tricks are possible to introduce a null model of the couplings by taking into account the effective strength of the couplings.

### 4 Experimental results

The proposed MEMM has a high flexibility due to the free specification on the parameters before making community assignments. In this section, we first verify that MEMM can correctly reflect our preference of the intra-layer edges by studying the performance of modularity and its modified form using different rewarding schemes for the parameters.
Then we discuss the coupling contribution in MEMM based on two rewarding schemes for the couplings. Finally the impact of the coupling strength $\varsigma$ and resolution parameter $\gamma_s$ is analyzed in 15 real-world multilayer networks to show that a better choice in practice is the one that obeys Definition 1.

For experimental purpose, we will take modularity as the objective function and utilize the well-known genLouvain method [22,31] for optimization. In the experiments, when it comes to inserting couplings between layers, we compute the Jaccard index of the neighbor sets of a specific physical node’s different copies

$$J(i, s, r) = \frac{\sum_k A_{iks} A_{ikr}}{\min(\sum_k A_{iks} + A_{ikr}, 1)}, \tag{9}$$

where matrix $A$ is the intra-layer adjacency matrix of the synthetic network. We then map the Jaccard index to a probability using a non-decreasing function (in this paper, linear functions are adopted) and sample couplings from this distribution.

### 4.1 Rewarding schemes

We begin by analyzing the influence of the rewarding scheme on the quality function. The proposed MEMM utilizes the $\lambda$ functions to distinguish the contribution type of the edges/couplings. Edges or couplings corresponding to the positive output of the $\lambda$ function are encouraged, i.e., a larger value of the corresponding parameter indicates a greater contribution. As discussed in Sect. 3.2, communities detected using modularity as global quality function will have a higher density of internal edges, which will lead to poor performance on the disassortative networks where edges are expected to be much more densely distributed between communities. Thus, the modularity is expected to provide poor detection result in a disassortative network due to the conflict about the definition of community. SBMs have been widely applied to successfully infer disassortative community structure, but it is less straightforward for modularity. Although some works have studied the relationship between minimizing modularity and disassortativity [20,29,34], it hasn’t been fully explored in the multilayer context. In particular, with the presence of couplings, directly minimizing multilayer modularity may not correctly find proper community assignments: the contribution of couplings takes totally opposite effect when minimizing the global objective function. In multilayer networks, a better choice is to reverse the contribution type of the intra-layer edges (while keeping the contribution type of the inter-layer couplings unchanged) in MEMM and obtain a similar form as Eq. (6), except the signs of $A_{ij}^s$ and $\gamma_s p_{ij}$ are flipped. This modified version would encourage the edges between the communities and discourage those within the communities since it has opposite preference of modularity within the layers.

To test the performance of the modularity and its modified form, we then generate two kinds of multilayer benchmark networks—a disassortative multilayer network whose layers are all disassortative networks and a assortative multilayer network whose communities have dense internal edges. In a disassortative network, the nodes are considered to belong to the same group only if there are no or very few edges lying between them. Thus the modularity is expected to provide poor detection result in this network due to the conflict definition of the “community”. However, the modified version of modularity aims exactly at such type of communities, so it is expected to exhibit better performance than the original version of modularity. Meanwhile, to highlight the difference of modifying rewarding scheme and direct minimization of modularity in the multilayer case, in what follows, we apply

---

1 The code is downloaded from [http://netwiki.amath.unc.edu/GenLouvain/GenLouvain](http://netwiki.amath.unc.edu/GenLouvain/GenLouvain).
such two methods upon the generated disassortative multilayer network. The results show that minimizing the multilayer modularity will lead to inconsistent community assignments among layers and the noises within each layer are fitted.

We first construct a single-layer disassortative network and then randomly flip the community label of a small amount of nodes to generate 10 noisy layers. For the single-layer disassortative network, we adopt the planted partition model [8], which is a generative model that constructs a network whose communities are random graphs. We divide 64 nodes into two communities with equal size (node 1 to node 32 and node 33 to node 64), and the edges between the nodes are sampled according to the corresponding linking probability. Here we assume the edges appear between the communities with probability $p_{out} = 0.4$ while that probability of the internal edges is $p_{in} = 0.005$. We then randomly choose up to 10 nodes and flip their community label to introduce noise. In this way, 10 noisy layers are generated and the couplings are added with probabilities proportional to the corresponding neighbor set similarity defined in Eq. (9).

For the assortative multilayer network, we adopt the Lancichinetti–Fortunato–Radicchi (LFR) benchmark networks [25], which extends the idea of the planted partition model and introduces power-law distribution to the degree of nodes and the edges between the communities. However, in this experiment we put no stress on the degree distribution due to the small network size. We generate 10 LFR benchmark networks as 10 layers, each of which consists of 64 nodes. These nodes have an average intra-layer degree 8 and are assigned to 4 communities with equal size and the community assignments are consistent among all layers. We insert exactly the couplings between the layers in the same manner as in the multilayer disassortative network to ensure that the major difference of these two multilayer network is the intra-layer edge distribution.

The detection results of the first layers with different rewarding schemes on these two networks are visualized in Fig. 2. Due to the space limit, here we compare the community structure discovered by the multilayer modularity and its modified version (i.e., with reversing edge contribution type) in the first layers of the two networks and similar analysis can be obtained in the remaining 9 layers of each network. As expected, modularity and the modified modularity have respective advantages on the corresponding network types. We first compare Fig. 2a and b, which illustrate the community assignment on the assortative network. The original modularity performs well on the networks where the community members are densely connected, as shown in Fig. 2a, while the modified version provides a poor assignment, as shown in Fig. 2b. From MEMM, we know the rewarding scheme adopted by the original modularity encourages the dense internal edges, which exactly matches the underlying community structure in a LFR benchmark. If we compare the detection result shown in Fig. 2c and d, the modified modularity outperforms that of the original modularity on the disassortative multilayer networks. The modified modularity adapts to the networks where community members are sparsely connected, which allows it to uncover communities with such high adhesion between communities, as illustrated in Fig. 2c. In contrast, the original modularity gives an unsatisfactory performance, as illustrated in Fig. 2d. Such distinguished preference of the original and modified modularity reflected in MEMM is the two strategies for encouraging and punishing the edges within the layers. For real-world networks, which usually takes an intermediate structure between the two networks considered here, we can then adopt an appropriate rewarding scheme and weighting the edges and couplings to meet the need.

Furthermore, the detection results obtained by maximizing the modified modularity and minimizing the original modularity in the disassortative network are shown in Fig. 3. There are $64 \times 10$ color blocks in both figures, corresponding to the nodes in the disassortative
Fig. 2 The community detection result of the first layers in multilayer assortative benchmark network and disassortative network, respectively. The subfigures a and b depict the detection results of the two methods on a assortative network. The subfigures c and d demonstrate the detection results of the two methods on a disassortative network. The nodes are spatially partitioned into groups according to the ground truth, i.e., nodes in the same community are placed close with each other. The detected community structures by the two methods are illustrated with different border color of each node, e.g., all nodes with purple border color are clustered into the same community by the algorithm in Fig. 2. a Modularity on the assortative multilayer network. b Modified modularity on the assortative multilayer network. c Modified modularity on the disassortative multilayer network. d Modularity on the disassortative network (color figure online)

multilayer network. Each color stands for a community, which may consist of nodes from any layers. We can see from Fig. 3a that there are two communities, each of which consists of nodes from every layer. Notice that the community assignments for each node obtained by maximizing the modified modularity match the ground-truth community label in each layer. However, if we just minimize the original modularity, there is no inter-layer community because of the penalty of couplings, as shown in Fig. 3b. Moreover, the noise in each layer now is fitted since there is no inter-layer information available. As a conclusion, minimizing the original multilayer modularity to find communities in disassortative networks is equivalent to processing the layers independently, while maximizing the modified multilayer modularity is able to reflect the dependence among different layers.

4.2 Coupling contribution

In the multilayer network model, as shown in Fig. 1, the couplings combine the layers to form a complex network structure. Thus the contribution of the couplings plays a crucial role in the correlation of the community assignments in different layers. According to Eq. (4),
The community detection result of the disassortative multilayer network. The subfigures a and b depict, respectively, the detection results obtained by maximizing the modified multilayer modularity and minimizing the original multilayer modularity. The network is constructed by adding noise to a disassortative single-layer network to generate 10 different layers, and then inserting couplings between node-copy pairs, see the text for details. There are $64 \times 10$ color blocks in both figures, corresponding to the nodes in the disassortative multilayer network. The color of the blocks represents the community label of the corresponding nodes. a By maximizing the modified multilayer modularity, we find two global communities, matching the ground-truth in the single-layer network used to generate the layers. The couplings lead to a consistent assignment for all the nodes across all the layers. b By minimizing the original multilayer modularity, we find that there are no communities consisting of nodes in different layers. Moreover, the noise is fitted in every layer, so that the assignments in each layer fail to match those in the single-layer network that generates them (color figure online).

...the coupling strength adopted by the multilayer modularity is equivalent to considering an equal contribution of the existing couplings while omitting the non-existing couplings. In this way, the contribution of the couplings takes value $\{0, \zeta\}$. If we further take the non-existing couplings into account, i.e., $\lambda_e(e_{isr}) = \lambda_h(h_{isr}) = -\lambda_f(f_{isr}) = \lambda_g(g_{isr}) = +\zeta$, the contribution of the couplings now takes $\{-\zeta, \zeta\}$. Notice that the coupling strength can make negative contribution once we take the non-existing couplings into consideration, which means the absence of couplings within a community and the presence of couplings between different communities will get punished. Therefore, whether we omit the non-existing couplings will influence the assignments of nodes in different layers. We then apply these two strategies on benchmark networks to examine the performance of them when the coupling structure varies.

To study the contribution of couplings to MEMM more comprehensively, we construct benchmark networks with different heterogeneity in the community structure of each layers and various coupling densities (i.e., the proportion of existing couplings in all possible couplings). More specifically, we construct 4 multilayer networks, respectively, with 0, 2, 3, 4 identical layers in them, to obtain 4 multilayer networks with different heterogeneity of the layers. We first generate 4 LFR benchmark networks with 128 nodes, 4 communities and an average intra-layer degree 16, and then duplicate one benchmark network to obtain several copies. Notice that the power-law nature of the LFR benchmark networks is not considered here due to the limited network size. The duplicates are chosen as the identical layers and the rest of the layers are chosen from the other three networks. We then insert couplings into these 4 layers by sampling from the probability proportional to the Jaccard index in descending order of $J(i, s, r)$ until the coupling density is $\rho$. We choose $\zeta = 6$ in this experiment to highlight the contribution of couplings. We will discuss about the impact...
Fig. 4 Tendency of the community detection accuracy in terms of NMI in different networks with different coupling density. The figures a, b, c, d are results for different degree of heterogeneity between layers. Each figure is illustrated as two wireframe meshes. In each figure, the left mesh is the result of ignoring the non-existing couplings and the right one is that of considering those couplings. Different networks (illustrated in each subfigure) consist of different number of identical layers to achieve different heterogeneity of the layers, as stated in corresponding captions. a 4 identical layers. b 3 identical layers. c 2 identical layers. d No identical layers.

of different coupling strength $\xi$ and the resolution parameter $\gamma_s$ in detail later in this section. As an indicator of the community detection quality, the normalized mutual information (NMI) [9,28] is calculated between the ground-truth labels and the obtained community assignments. NMI measures the similarity between two given vectors and is widely adopted to evaluate the accuracy of a partition, where a higher value indicates a better partition result.

We apply the two strategies of coupling contribution to these networks and illustrate the corresponding NMI for each layer in different settings, as demonstrated in Fig. 4. The detection results of modularity ignoring and considering the non-existing couplings are shown as the left and right mesh, respectively, in each subfigure, and the degree of heterogeneity of layers are indicated by the number of identical layers. From Fig. 4a we can see, when the layers are all the same, the two strategies both lead to a good assignment. However, as the heterogeneity increases (i.e., as the number of identical layers decreases from 4 to 0), the quality functions exhibit gradually worse detection results on networks with high coupling density, as shown from Fig. 4a–d. This suggests that the couplings will force the layers to make a similar community assignment to the nodes, which results in failing to discover the heterogeneity of layers. The similar layers will greatly influence the assignments of the other layers. For example, in Fig. 4b, when there are two layers that have similar community structure, the modularity will guide us to find the first two layers which are identical. The community structures in the rest two layers are omitted due to a high coupling density.

What is more, by comparing the NMI of the two strategies, we find that considering the non-existing couplings performs better as the coupling density $\rho$ and the heterogeneity
of layers increase, except when the coupling density $\rho$ is around 0.5. This confirms that considering the non-existing couplings in MEMM will be more likely to make a balanced assignment when the layers show great heterogeneity and the couplings are dense. The decline of performance at $\rho = 0.5$ when considering the non-existing couplings in modularity actually arises due to the optimization method we used. In fact, if we denote the number of the internal and external existing couplings as $C_I = \sum_{i=1}^{l} C_{isr} \delta(u_{is}, v_{ir})$ and $C_E = \sum_{i=1}^{l} C_{isr} (1 - \delta(u_{is}, v_{ir}))$, and the non-existing couplings as $C_I = \sum_{i=1}^{l} (1 - C_{isr}) \delta(u_{is}, v_{ir})$ and $C_E = \sum_{i=1}^{l} (1 - C_{isr}) (1 - \delta(u_{is}, v_{ir}))$, we can rewrite the contribution of the couplings (considering the non-existing couplings) as

$$\mathcal{M}_{\text{coupling}} = \varsigma(C_I - C_I - C_E + C_E) = \varsigma[2(C_I + C_E) - M],$$

(10)

where $M = C_I + C_I + C_E + C_E = l(l - 1)N$ is the number of all possible couplings in the network, $N$ is the number of nodes within a layer and $l$ is the number of layers. Here, the number of couplings of each type $C_x$ depends on the coupling density $\rho$ and the current community assignment.

We can see that the contribution of the couplings depends on the number of couplings present within the communities and absent between different communities, fixing the coupling strength. As $\rho$ increases, $\mathcal{M}_{\text{coupling}}$ goes to $\varsigma(2M_I - M)$, where $M_I = C_I |_{\rho=1} = C_j |_{\rho=0}$ is the extreme number of internal couplings. This indicates that the contribution is governed by the total amount of internal couplings. Similarly, if $\rho \to 0$ the contribution will be governed by the external non-existing couplings with maximum $\varsigma(2M_E - M)$, where $M_E = C_E |_{\rho=1} = C_E |_{\rho=0}$. When $\rho$ varies, we have

$$\frac{\mathcal{M}_{\text{coupling}}}{\varsigma} = 2[\rho M_I + (1 - \rho)M_E] - M = 2[\rho (M - M_E) + (1 - \rho)M_E] - M = (2\rho - 1)(M_E - M_I),$$

(11)

given the community assignment. The quantities $M_E$ and $M_I$ are functions of the community assignment. Thus the contribution of the couplings is composed of the strength parameter $\varsigma$, the current community assignment and the coupling strength. Equation (11) suggests that the contribution of couplings is expected to be positive if the coupling density is high and there are more internal couplings $M_I$ than external couplings $M_E$, or low $\rho$ with more external couplings. In particular, the contribution is expected to vanish when the coupling density $\rho$ goes to 0.5. At this time, the Louvain method will generate a less optimal assignment owing to the heuristically merge. The global contribution of the couplings is expected to be zero, but when the Louvain method attempts to merge two communities to locally increase the modularity, it will take the couplings into consideration. Thus the error is accumulated during the iteration and finally results in a less satisfactory result.

In contrast, if we omit the terms of non-existing couplings, we obtain a different representation of the contribution:

$$\frac{\mathcal{M}_{\text{coupling}}}{\varsigma} = C_I - C_E = \rho (M_I - M_E).$$

(12)

It will not suffer the problem of contribution vanish as it is a linear function of $\rho$. However, the influence of couplings increases as $\rho$ increases, to force the layers to adopt a similar assignment. As a result, the heterogeneity of layers is erased.
Table 1 The 15 multilayer network data and respective statistics used in the experiment

| Network                                      | #Nodes | #Layers |
|----------------------------------------------|--------|---------|
| Bos multiplex genetic                        | 325    | 4       |
| Candida multiplex genetic                    | 367    | 7       |
| Celegans multiplex genetic                  | 3879   | 6       |
| Celegans multiplex neuronal                 | 279    | 3       |
| CKM-physicians-innovation multiplex Social   | 246    | 3       |
| EUAir multiplex transport                   | 450    | 37      |
| FAO multiplex trade                          | 214    | 364     |
| HumanHerpes4 multiplex genetic              | 216    | 4       |
| HumanHIV1 multiplex genetic                 | 1005   | 5       |
| London multiplex transport                  | 369    | 3       |
| PierreAuger multiplex coauthorship          | 514    | 16      |
| Plasmodium multiplex genetic                | 1203   | 3       |
| Rattus multiplex genetic                    | 2640   | 6       |
| SacchPomb multiplex genetic                 | 4092   | 7       |
| Xenopus multiplex genetic                   | 461    | 5       |

The data are available at http://deim.urv.cat/~manlio.dedomenico/data.php

In a nutshell, considering the non-existing couplings helps fight against the decline of performance when the degree of heterogeneity of layers is large and coupling density is high. However, it will suffer slight performance decline when the coupling density is around 0.5 when using a local heuristic optimization strategy. In contrast, ignoring the coupling density makes the quality function less robust to the high coupling density networks, but the contribution of couplings does not vanish so it will not suffer from the performance decline with heuristic methods.

4.3 Parameter analysis

By deriving the multilayer modularity using MEMM, two parameters $\gamma_s$ and $\zeta$ are introduced to control the behavior of the quality function. The resolution parameter $\gamma_s$ controls the size of the detected communities, and the coupling strength $\zeta$ determines the contribution of couplings [31]. Nevertheless, if we interpret the parameters of modularity from the perspective of MEMM, varying these parameters signifies violation of Definition 1. To study the impact on the detection results when violating the discriminative condition in choosing parameters, we apply the multilayer modularity with different parameter settings in 15 real-world multilayer networks, as listed in Table 1. For each network, we insert couplings to all possible node pairs between adjacent layers. We adopt the genLouvain method for optimization, and for each network, 10 runs will be conducted and the average results are reported to reduce the random behavior introduced by the genLouvain method. We repeat the detection process with $\zeta = \{0.1, 0.2, \ldots, 1.5\}$ while keeping $\gamma_s = 1$ and $\gamma_s = \{0.1, 0.2, \ldots, 1.5\}$ while keeping $\zeta = 1$, respectively, and differentiate the contribution of inter-layer couplings and intra-layer edges to the modularity with different colors. For simplicity, we make $\gamma_s$ takes the same value for all layers and omit the absent couplings between layers. Since there is no ground-truth community labels in these real-world multilayer networks, the result is evaluated in terms of the modularity value in Eq. (6), leaving out the constant values which does not affect the
Fig. 5 Parameter analysis on $\gamma_s$: The detection results in the 15 real multilayer (multiplex) networks using modularity as the quality function, with $\zeta = 1$ fixed and $\gamma_s = \{0.1, 0.2, \ldots, 1.5\}$. The two color of the stacked bar in each figure indicates the contribution of couplings and edges as the resolution limit $\gamma_s$ varies. Here $\gamma_s$ takes the same value in all layers. The polyline demonstrates the change of the number of communities. Please see the main text for detailed analysis of the results. a Bos multiplex genetic. b Candida multiplex genetic. c Celegans multiplex genetic. d Celegans multiplex neuronal. e CKM-physicians-innovation multiplex social. f EUAir multiplex transport. g FAO multiplex trade. h HumanHerpes4 multiplex genetic. i HumanHIV1 multiplex genetic. j London multiplex transport. k PierreAuger multiplex coauthorship. l Plasmodium multiplex genetic. m Rattus multiplex genetic. n SacchPomb multiplex genetic. o Xenopus multiplex genetic (color figure online).
optimization. Note that when $\zeta \to \infty$, the modularity goes to infinity even if the partition is unchanged, so we divide the contribution of couplings by $\zeta$ for normalization. Since the contribution difference between the inter-layer couplings and intra-layer edges can be too large (mostly owing to the high density of inter-layer couplings in this experiment), we adjust the proportion of the two edge types to highlight their contribution changes. The experimental results will confirm why the parameter setting satisfying Definition 1, i.e., $\gamma_s = 1$, $\zeta = 1$, is a relatively reliable choice in general.

Figure 5 plots the parameter analysis results on $\gamma_s$, i.e., the trend of the modularity contribution and the total number of communities detected by varying $\gamma_s = \{0.1, 0.2, \ldots, 1.5\}$ and fixing $\zeta = 1$. Notice that given the total number of nodes in each layer, the average size of communities is inversely proportional to the number of communities. As $\gamma_s$ increases, the contribution of the intra-layer edges decreases significantly. In contrast, the contribution of couplings stays the same since the community assignments of each node are consistent across all layers. When $\gamma_s = 0$, there is no punishment to the non-existing internal edges, so the community can be as large as possible. As a consequence, all nodes in the layer will be assigned into the same community. Here we do not plot the case where $\gamma_s = 0$ due to dramatic increase in the total number of communities from $\gamma_s = 0$ to $\gamma_s = 0.1$ in some networks. When $\gamma_s \to \infty$, grouping any non-clique (i.e., a subgraph in which there are at least two nodes that are not connected by an edge) will lead to intolerable punishment, so that the network is partitioned into isolated cliques, with no edges connecting them. Therefore, the modularity value has a lower bound which corresponds to the partition of such isolated cliques. As $\gamma_s$ decreases, the sizes of detected communities shrink by splitting into smaller ones until reaching the lower bound. Overall, we can see that the decreasing speed of modularity slows down, which indicates that the partition within each layer becomes more stable. In addition, when no priori knowledge of community size is available, choosing a moderate value for $\gamma_s$ will produce moderate-sized communities. In this experiment, we find that around $\gamma_s = 1$, the decreasing speed of modularity value is relatively slow and the total number of communities found is moderate among these parameter settings. Therefore, $\gamma_s = 1$ is a reasonable parameter setting when we have no insights into the network data.

Figure 6 plots the parameter analysis results on $\zeta$, i.e., the trend of the detection results as $\zeta$ varies in $\{0.1, 0.2, \ldots, 1.5\}$, with $\gamma_s = 1$ fixed for all layers in the 15 real multilayer networks. Note that here the contribution of couplings is normalized by dividing $\zeta$. The contribution of couplings grows as the coupling strength $\zeta$ grows, until a maxima is reached. This upper bound corresponds to the case where the nodes are assigned to the same community as their counterparts in other layers. Overall, we find such upper bound is reached when $\zeta = 1$ for all networks in this experiment. Notice that here the coupling density is 1 for all of the networks. Therefore, when we need to introduce heterogeneity into the community assignments of different layers, a common practice is to start from $\zeta = 1$ and reduce its value accordingly.
Otherwise when considering a consistent assignment for nodes in different layers, \( \zeta = 1 \) is a reliable choice.

We have analyzed the impact of different parameter settings in the 15 real-world multilayer networks using modularity as the quality function in this section. The parameter \( \gamma_s \) controls the community scale within each layer and \( \zeta \) controls the consistency of community assignments across the layers. In a nutshell, when there is no priori knowledge available, choosing \( \gamma_s = 1 \) and \( \zeta = 1 \) will lead to moderate-sized communities which is highly consistent across layers. Based on the detection result using such parameter setting, one is able to adjust the parameters in order to discover communities that satisfies practical needs.

### 5 Conclusion

In this paper, we presented the multilayer edge mixture model (MEMM) as the hyper model to provide a new interpretation of a variety of the existing multilayer community structure quality functions and pave the way for the derivation of new ones. The proposed model is based on a mixture of the edge contributions, which explores a relatively general representation of the community structure quality functions. As the main example in this paper, the multilayer modularity has been derived from the proposed model. We studied how the multilayer edge mixture model evaluates the community structure by generating a quality function according to the definition of the community, and discussed how to choose a discriminative parameters set for balancing the contributions.

A promising direction of further studying MEMM is to comparing the specific weighting function adopted by different community structure quality functions. This will deepen our understanding about the exact difference of these quality functions and help better interpreting their distinguished performance on different network types. Moreover, it may also be possible for us to combine the advantages of different quality functions. Another important extension is to enable MEMM to represent directed networks and weighted networks. With these features, MEMM may become a generic community quality function analyzing tool, which is exactly what we lack in the literature.

**Acknowledgements** This work was supported by NSFC (61502543 and 61672313), National Key Research and Development Program of China (2016YFB1001003), Guangdong Natural Science Funds for Distinguished Young Scholar (2016A030306014), Tip-top Scientific and Technical Innovative Youth Talents of Guangdong special support program (2016TQ03X542), NSF through Grants IIS-1526499, IIS-1763325, and CNS-1626432.

**References**

1. Bang-Jensen J, Gutin GZ (2008) Digraphs: theory, algorithms and applications. Springer, Berlin
2. Barrat A, Barthelemy M, Pastor-Satorras R, Vespignani A (2004) The architecture of complex weighted networks. Proc Natl Acad Sci USA 101(11):3747–3752
3. Bazzi M, Porter MA, Williams S, McDonald M, Fenn DJ, Howison SD (2014) Community detection in temporal multilayer networks, and its application to correlation networks, arXiv preprint arXiv:1501.00040
4. Boccaletti S, Bianconi G, Criado R, Del Genio CI, Gómez-Gardeñes J, Romance M, Sendina-Nadal I, Wang Z, Zanin M (2014) The structure and dynamics of multilayer networks. Phys Rep 544(1):1–122
5. Bollobás B (1998) Modern graph theory, vol 184. Springer, Berlin
6. Cardillo A, Gómez-Gardeñes J, Zanin M, Romance M, Papo D, del Pozo F, Boccaletti S (2013) Emergence of network features from multiplexity. Sci Rep 3 https://doi.org/10.1038/srep01344
7. Clauset A, Newman ME, Moore C (2004) Finding community structure in very large networks. Phys Rev E 70(6):066111
8. Condon A, Karp RM (2001) Algorithms for graph partitioning on the planted partition model. Random Struct Algorithms 18(2):116–140
9. Danon L, Diaz-Guilera A, Duch J, Arenas A (2005) Comparing community structure identification. J Stat Mech: Theory Exp 09:P09008
10. De Domenico M, Lancichinetti A, Arenas A, Rosvall M (2015) Identifying modular flows on multilayer networks reveals highly overlapping organization in interconnected systems. Phys Rev X 5(1):011027
11. Delvenne J-C, Yaliraki SN, Barahona M (2010) Stability of graph communities across time scales. Proc Natl Acad Sci 107(29):12755–12760
12. Djidjev HN (2006) A scalable multilevel algorithm for graph clustering and community structure detection. Algorithms and models for the web-graph. Springer, Berlin pp 117–128
13. Doreian P, Mrvar A (2009) Partitioning signed social networks. Soc Netw 31(1):1–11
14. Fortunato S (2010) Community detection in graphs. Phys Rep 486(3):75–174
15. Gao J, Buldyrev SV, Havlin S, Stanley HE (2012) Robustness of a network formed by n interdependent networks with a one-to-one correspondence of dependent nodes. Phys Rev E 85(6):066134
16. Girvan M, Newman ME (2002) Community structure in social and biological networks. Proc Natl Acad Sci 99(12):7821–7826
17. Guimerà R, Sales-Pardo M (2009) Missing and spurious interactions and the reconstruction of complex networks. Proc Natl Acad Sci 106(52):22073–22078
18. Holland PW, Laskey KB, Leinhardt S (1983) Stochastic blockmodels: first steps. Soc Netw 5(2):109–137
19. Holme P, Saramäki J (2012) Temporal networks. Phys Rep 519(3):97–125
20. Hu H-B, Wang X-F (2009) Disassortative mixing in online social networks. Europhys Lett 86(1):18003
21. Hu Y, Li M, Zhang P, Fan Y, Di Z (2008) Community detection by signaling on complex networks. Phys Rev E 78(1):016115
22. Jeub LGS, Bazzi M, Jutla IS, Mucha PJ (2011-2017) A generalized louvain method for community detection implemented in matlab. http://netwiki.amath.unc.edu/GenLouvain
23. Karrer B, Newman ME (2011) Stochastic blockmodels and community structure in networks. Phys Rev E 83(1):016107
24. Kivell M, Arenas A, Barthelemy M, Gleeson JP, Moreno Y, Porter MA (2014) Multilayer networks. J Complex Netw 2(3):203–271
25. Lancichinetti A, Fortunato S, Radicchi F (2008) Benchmark graphs for testing community detection algorithms. Phys Rev E 78(4):046110
26. Li J-H, Wang C-D, Li P-Z, Lai J-H (2018) Discriminative metric learning for multi-view graph partitioning. Pattern Recognit 75:199–213
27. Li W, Liu C-C, Zhang T, Li H, Waterman MS, Zhou XJ (2011) Integrative analysis of many weighted co-expression networks using tensor computation. PLoS Comput Biol 7(6):e1001106
28. Meila M (2007) Comparing clusterings—an information based distance. J Multivar Anal 98(5):873–895
29. Mieghem PV, Ge X, Schumm P, Trajanovski S, Wang H (2010) Spectral graph analysis of modularity and assortativity. Phys Rev E 82:056113
30. Moosavi SA, Jalali M, Misaghian N, Shamsirband S, Anisi MH (2016) Community detection in social networks using user frequent pattern mining. Knowl Inf Syst. https://doi.org/10.1007/s10115-016-0970-8
31. Mucha PJ, Richardson T, Macon K, Porter MA, Onnela J-P (2010) Community structure in time-dependent, multiscale, and multiplex networks. Science 328(5980):876–878
32. Newman M (2010) Networks: an introduction. Oxford University Press, Oxford
33. Newman ME (2004) Analysis of weighted networks. Phys Rev E 70(5):056131
34. Newman ME (2006) Modularity and community structure in networks. Proc Natl Acad Sci 103(23):8577–8582
35. Newman ME, Girvan M (2004) Finding and evaluating community structure in networks. Phys Rev E 69(2):026113
36. Peixoto TP (2015) Inferring the mesoscale structure of layered, edge-valued, and time-varying networks. Phys Rev E 92(4):042807
37. Radicchi F, Castellano C, Cecconi F, Loreto V, Parisi D (2004) Defining and identifying community members in networks. Proc Natl Acad Sci USA 101(9):2658–2663
38. Reichardt J, Bornholdt S (2006) Statistical mechanics of community detection. Phys Rev E 74(1):016110
39. Reichardt J, White DR (2007) Role models for complex networks. Eur Phys J B 60(2):217–224
40. Rocklin M, Pinar A (2013) On clustering on graphs with multiple edge types. Internet Math 9(1):82–112
41. Rosvall M, Bergstrom CT (2007) An information-theoretic framework for resolving community structure in complex networks. Proc Natl Acad Sci 104(18):7327–7331
42. Rosvall M, Bergstrom CT (2008) Maps of random walks on complex networks reveal community structure. Proc Natl Acad Sci 105(4):1118–1123
43. Strogatz SH (2001) Exploring complex networks. Nature 410(6825):268–276
44. Szell M, Lambiotte R, Thurner S (2010) Multirelational organization of large-scale social networks in an online world. Proc Natl Acad Sci 107(31):13636–13641
45. Verbrugge LM (1979) Multiplexity in adult friendships. Soc Forces 57(4):1286–1309
46. Wang CD, Lai JH, Yu PS (2014) NEIWalk: community discovery in dynamic content-based networks. IEEE Trans Knowl Data Eng 26(7):1734–1748
47. Wang Y, Xun J, Yang Z, Li J (2017) Query optimal k-plex based community in graphs. Data Sci Eng 2:257
48. Wasserman S, Faust K (1994) Social network analysis: methods and applications, vol 8. Cambridge University Press, Cambridge
49. Yang B, Cheung WK, Liu J (2007) Community mining from signed social networks. IEEE Trans Knowl Data Eng 19(10):1333–1348
50. Yang J, Leskovec J (2015) Defining and evaluating network communities based on ground-truth. Knowl Inf Syst 42(1):181–213. https://doi.org/10.1007/s10115-013-0693-z
51. Zhang S, Wang R-S, Zhang X-S (2007) Uncovering fuzzy community structure in complex networks. Phys Rev E 76(4):046103

Han Zhang received the B.S. degree in Software Engineering in 2017 from Sun Yat-sen University, Guangzhou, China. He is currently pursuing his Ph.D. degree from Tsinghua University, Beijing, China. His research interest is machine learning and data mining.

Chang-Dong Wang received the B.S. degree in applied mathematics in 2008, the M.Sc. degree in computer science in 2010, and the Ph.D. degree in computer science in 2013, all from Sun Yat-sen University, Guangzhou, China. He was a visiting student at the University of Illinois at Chicago from January 2012 to November 2012. He is currently an Associate Professor with the School of Data and Computer Science, Sun Yat-sen University, Guangzhou, China. His current research interests include machine learning and data mining. He has published more than 90 scientific papers in international journals and conferences such as IEEE TPAMI, IEEE TKDE, IEEE TCYB, IEEE TSMC-C, Pattern Recognition, KAIS, Neurocomputing, ICDM, CIKM and SDM. His ICDM 2010 paper won the Honorable Mention for Best Research Paper Awards. He was awarded 2015 Chinese Association for Artificial Intelligence (CAAI) Outstanding Dissertation.
Jian-Huang Lai received the M.Sc. degree in applied mathematics in 1989 and the Ph.D. degree in mathematics in 1999 from Sun Yat-sen University, China. He joined Sun Yat-sen University in 1989 as an Assistant Professor, where he is currently a Professor with the School of Data and Computer Science. His current research interests include the areas of digital image processing, pattern recognition, multimedia communication, wavelet and its applications. He has published more than 200 scientific papers in the international journals and conferences on image processing and pattern recognition, such as IEEE TPAMI, IEEE TKDE, IEEE TNN, IEEE TIP, IEEE TSMC-B, Pattern Recognition, ICCV, CVPR, IJCAI, ICDM and SDM. Prof. Lai serves as a Standing Member of the Image and Graphics Association of China, and also serves as a Standing Director of the Image and Graphics Association of Guangdong.

Philip S. Yu received the B.S. degree in E.E. from National Taiwan University, the M.S. and Ph.D. degrees in E.E. from Stanford University, and the M.B.A. degree from New York University. He is a Distinguished Professor in Computer Science at the University of Illinois at Chicago and also holds the Wexler Chair in Information Technology. Before joining UIC, Dr. Yu was at IBM TJ Watson Research Center, where he was manager of the Software Tools and Techniques group. His research interest is on big data, including data mining, data stream, database and privacy. He has published more than 810 papers in refereed journals and conferences. He holds or has applied for more than 300 US patents. Dr. Yu is a Fellow of the ACM and the IEEE. He is the recipient of ACM SIGKDD 2016 Innovation Award for his influential research and scientific contributions on mining, fusion and anonymization of big data, the IEEE Computer Society’s 2013 Technical Achievement Award for “pioneering and fundamentally innovative contributions to scalable indexing, querying, searching, mining and anonymization of big data”, and the Research Contributions Award from IEEE Intl. Conference on Data Mining (ICDM) in 2003 for his pioneering contributions to the field of data mining. He also received an IEEE Region 1 Award for “promoting and perpetuating numerous new electrical engineering concepts” in 1999. He had received several UIC honors, including Research of the Year at 2013 and UI Faculty Scholar at 2014. He also received many IBM honors including 2 IBM Outstanding Innovation Awards, an Outstanding Technical Achievement Award, 2 Research Division Awards and the 94th plateau of Invention Achievement Awards. He was an IBM Master Inventor.