Fusion of Community Structures in Multiplex Networks by Label Constraints

Yuming Huang
Dept. of Physics
North Carolina State University
Raleigh, NC, USA

Ashkan Panahi, Hamid Krim
Dept. of Electrical and Computer Engineering
North Carolina State University
Raleigh, NC, USA

Liyi Dai
Computing Sciences Division
U.S. Army Research Office
Durham, NC, USA

Abstract—**We develop a Belief Propagation algorithm for community detection problem in multiplex networks, which more accurately represents many real-world systems. Previous works have established that real world multiplex networks exhibit redundant structures/communities, and that community detection performance improves by aggregating (fusing) redundant layers which are generated from the same Stochastic Block Model (SBM). We introduce a probability model for generic multiplex networks, aiming to fuse community structure across layers, without assuming or seeking the same SBM generative model for different layers. Numerical experiment shows that our model finds out consistent communities between layers and yields a significant detectability improvement over the single layer architecture. Our model also achieves a comparable performance to a reference model where we assume consistent communities in prior. Finally we compare our method with multilayer modularity optimization in heterogeneous networks, and show that our method detects correct community labels more reliably.**

Index Terms—**Community detection, Multiplex network, Fusion, Belief propagation, SBM**

I. Introduction

Multiplex networks have been studied and applied to model many real world complex systems [1]–[6]. With different connectivities in different layers, multiplex networks allow for richer structure and dynamics compared to single layer networks. Community detection in multiplex networks has been receiving increasing attention in the recent literature [7]–[15]. Due to the variety of multiplex structures, these studies address different assumptions and circumstances. Community detection is essential in a wide range of applications, such as temporal brain networks [16], social networks [17], [18], etc.

In the content of community detection, multiplex networks have advantages in representing community structures over single layer networks. For example, when a node belongs to multiple communities, it is easier to retain the community structure using different layers rather than overlapping them in a single layer [18]. Multiplex networks can also model time-varying communities, by encoding community structure at a certain time point in each layer [16]. Domenico et al. [5] found that many real world multiplex networks exhibit redundant layers and introduced an information theoretic measure to identify them. A natural idea is to take advantages of redundant communities across layers, by treating them as multiple realizations of the same community and fusing them to increase its ”signal-to-noise ratio”. Taylor et al. [11] studied a special case where different layers have exactly the same community structure, i.e. the layers are generated from the same stochastic block model (SBM). They showed that by simply aggergating edges from different layers into one layer, the community detectability is increased. Stanley et al. [12] considered multilayer networks generated by multiple SBMs and each SBM corresponds to a set of redundant layers. The authors first estimate SBM parameters for each layer and cluster similar ones. However, this model still requires community structures in a layer being completely consistent with another layer. A more generic model allows for heterogeneous community structures in different layers. In this paper, we introduce a generative model with the mere constraint that a community label refers to a constant set of nodes. Without seeking layers with consistent SBM, or assuming consistent community structures as prior knowledge, our model finds out when a community appears in different layers and fuse them, hence improving community detectability.

Decelle et al. recently developed a Belief Propagation algorithm to solve community detection from a single layer in an asymptotically optimal way [19], [20], by viewing the community detection as a Bayesian inference problem of SBM parameters. Since no other algorithm will have a better performance, the detection accuracy of their method indicates the inherent detectability of a community structure. Our multiplex model aims to enhance the detectability of repeated communities in multiple layers. We develop a Bayesian model for our generic multiplex networks and solve it by extending Belief Propagation equations in [19]. Thus the performance of our algorithm indicates the community detectability given a set of mild constraints (refer to Well Partitioning Property in Sec. III). As a comparison, we designed a reference model with stronger constraints that all layers have consistent community structure. Our main contribution is in showing that despite the community consistency constraint between layers, one can still achieve a similar community detection performance improvement over single layer. This is realized by merely exploiting some natural properties of real-world communities, specifically, consistent node membership of a certain community. An additional advantage of our method

We would like to acknowledge the support of U.S. Army Research Office: Grant # W911NF-16-2-0005

over layer aggregation [11], [12], is that it does not require layers to be generated from the same SBM. For heterogeneous multiplex networks, where community structure of an entire layer does not completely match another layer, our method is still straightforward to apply. Our method is able to identify and fuse the part of a multiplex network corresponding to consistent communities, thus improving detectability.

II. Problem Statement

A multiplex network $W = (V, E(1), E(2), ..., E(L))$ is composed of a set of nodes $V = \{v_1, v_2, ..., v_N\}$ and a set of edges $E(\cdot)$ for each of the L layers. The goal is to identify a set of communities $C = \{C_1, C_2, ..., C_q\}$, in which $C_i \subseteq V$ is a set of densely connected nodes, i.e. community, in at least one layer. Each layer has a non-exclusive subset of C and different layers may have common communities. Common communities are to be fused to improve community detectability. For simplicity, we assume each layer can be described by an ordinary SBM, which means communities in the same layer do not overlap (have common nodes). This nevertheless allows for different communities in different layers to overlap.

III. Multiplex SBM and WPP

Stochastic Block Model (SBM) has been a popular generative model for community structures in networks [21]. A community corresponds to a dense block on the diagonal in the adjacency matrix. SBM generates a network from a set of parameters: the number of communities q, the size of each community $\{n_a\}$, the affinity matrix $\{p_{ab}\}$ and community assignments $t_i \in \{1, ..., q\}$. To generate a network layer, nodes are connected by a probability of $\{p_{ab}\}$, where a and b stand for the community labels of the nodes. Community detection can be understood as an inference problem of SBM parameters given the network.

We introduce a multiplex SBM which generates the multiplex network described in the problem statement. In our model, SBM in each layer works the same way as in a single layer network, but they are associated with each other by constraints and potential common communities. We describe the constraint by a Well Partitioning Property (WPP). This property requires that a community C_a refers to the same set of nodes if it exists in multiple layers, and it does not allow overlapping community in any layer. If we denote the set of adjcancy matrices as $\{W(l)\}$ and the set of community label vectors on nodes in all layers as $\{\boldsymbol{t}(l)\}$, we propose the generative probability model of a multiplex network with some community structure given a set of parameters, as follows,

$$\begin{aligned} &P(\{W(l)\}, \{\boldsymbol{t}(l)\}|p, q, \{n_a\}) \\ &= \frac{1}{Z} \prod_{(i,j),(l,l')} f_{check}(t_i(l), t_j(l), t_i(l'), t_j(l')) \\ &\times \prod_{l=1}^{L} \left[\prod_{(i,j)\in E(l)} p_{t_i(l),t_j(l)} \prod_{(i,j)\notin E(l)} (1 - p_{t_i(l),t_j(l)}) \prod_i n_{t_i(l)} \right]. \end{aligned} \tag{1}$$

Here Z is a normalization constant, and the second row on the right hand side shows the multiplication of single-layer SBM formulation [19] in all L layers, while the f_{check} functional sets the local constraint of community labels between any two layers l and l'. By checking over every pair of nodes, it ensures WPP is satisfied globally. In particular, $f_{check} = 1$ implies that the labels satisfy the following condition and $f_{check} = 0$ otherwise.

$$\text{Assume } \begin{cases} t_i(l) = \alpha \\ t_j(l) = \beta \end{cases},$$

$$\text{If } \alpha = \beta, \text{ then } \begin{cases} t_i(l') = t_j(l') = \alpha \\ \text{or } \begin{cases} t_i(l') \neq \alpha \\ t_j(l') \neq \alpha \end{cases} \end{cases},$$

$$\text{If } \alpha \neq \beta, \text{ then } \begin{cases} t_i(l') \neq \beta \\ t_j(l') \neq \alpha \end{cases}.$$

This set of conditions avoid all possible situations that community labels of copies of two nodes in two layers violate WPP locally. Here we give a simple example to explain why WPP enables the model to find out consistent communities across layers and thus improve detection performance.

Assuming there are q communities in a two layer network and only one of them is present in both layers (we refer to them as two identical clusters), the task is to assign q different labels to $q+1$ clusters. The ideal result is that the two clusters representing the same community are assigned the same label. However, if we assign them different labels, there will not be enough labels for the rest of the clusters. That will result in two different clusters being identified as the same community, which violates WPP and cause f_{check} function to have 0 value for some nodes. Hence, only by assigning the two identical clusters the same community label, can we obtain a non-zero probability in Eq. (1).

IV. Belief Propagation Algorithm

Belief Propagation is known as an asymptotically optimal algorithm to solve the community detection problem for networks described by stochastic block models [19], [20]. The nature of a Belief Propagation algorithm allows for parallel or distributed implementation of efficient inference for large networks. We use a factor graph [22] to represent the network model and the WPP condition in Eq. (1). A factor graph is composed of factor nodes representing different terms in Eq. (1) (such as f_{check} and p_{ab}) and variable nodes (such as an actual node in a layer in the multiplex network). A factor node connects to the variables appearing in the argument of its corresponding term in Eq. (1). The resulting factor graph allows us to utilize the well-known Belief Propagation algorithm to perform node-community inference. Within each layer, the factor graph structure is the same as the one for single layer explained in [19], where factor nodes define pairwise interaction between nodes about establishing their connectivity by an edge in the network. However between different layers, there are factor nodes associated with the constraint function f_{check} between copies of two nodes in two layers, amounting to one factor node connecting to four different variable nodes. In this case, we explicitly write the messages between those variable nodes and factor nodes. Then using the sum-product rule [22], we can write update equations for the messages as follows:

Intra-layer message:

$$\begin{aligned} m_{t_i}^{i\to a}(l) &= \frac{1}{Z^{i\to a}(l)} n_{t_i(l)} e^{-h_{t_i(l)}} \\ &\prod_{d\in N_{intra}(i(l))\backslash a} \left(\sum_{t_d} c_{t_d t_i(l)} m_{t_d}^{d\to i(l)} \right) \\ &\times \prod_{c\in N_{inter}(i)} \left(\sum_{t_j(l),t_i(l'),t_j(l')} f_{check}(t_i(l),t_i(l'),t_j(l),t_j(l')) \right. \\ &\left. \prod_{k\in N_{inter}(c)\backslash i} m_{t_k}^{k\to c} \right). \end{aligned} \tag{2}$$

Inter-layer message:

$$\begin{aligned} m_{t_i(l)}^{i\to c} &= \frac{1}{Z^{i\to c}(l)} n_{t_i(l)} e^{-h_{t_i(l)}} \prod_{d\in N_{intra}(i)} \left(\sum_{t_d} c_{t_d t_i} m_{t_d}^{d\to i} \right) \times \\ &\prod_{c^*\in N_{inter}(i)\backslash c} \left(\sum_{t_j(l),t_i(l'),t_j(l')} f_{check^*}(t_i(l),t_i(l'),t_j(l),t_j(l')) \right. \\ &\left. \prod_{k\in N_{inter}(c^*)\backslash i} m_{t_k}^{k\to c^*} \right), \end{aligned} \tag{3}$$

where $N_{intra}(i)$ represents intra-layer neighbors of node i, and $N_{inter}(i)$, represents those of the inter-layer ones (the constraint-checking factors). l and l' are indices of layers connected by the constraint-checking factor c. $c_{ab} = Np_{ab}$ where N is the number of nodes, and $h_{t_i} = \frac{1}{N}\sum_k \sum_{t_k} c_{t_k t_i} b_{t_k}^k$ is an external field to approximate influences from unconnected nodes [19].

The belief $b_{t_i}^i(l)$ of a node in layer l is its marginal distribution of community labels. Each node belief is calculated by:

$$\begin{aligned} b_{t_i}^i(l) &= \frac{1}{Z^i(l)} n_{t_i(l)} e^{-h_{t_i(l)}} \prod_{d\in N_{intra}(i(l))} \left(\sum_{t_d} c_{t_d t_i(l)} m_{t_d}^{d\to i(l)} \right) \\ &\times \prod_{c\in N_{inter}(i(l))} \left(\sum_{t_j(l),t_i(l'),t_j(l')} f_{check}(t_i(l),t_i(l'),t_j(l),t_j(l')) \right. \\ &\left. \prod_{k\in N_{inter}(c)\backslash i(l)} m_{t_k}^{k\to c} \right). \end{aligned} \tag{4}$$

In our implementation, the algorithm randomly initializes all messages and applies message update equations iteratively. In each epoch, we make sure each message is updated once, and the order of updates is arbitrary. After sufficient number of iterations, if communities are detectable, the messages will converge to a fixed point. We subsequently calculate the belief of each node, which are in turn used to pick a community label.

V. Numerical Experiment

As an evaluation, we compare the performance of our model on a two layer network with the original model [19] on a single layer network. We also design a reference model with a stronger prior that the two layers have the same community structures. Similar to layer aggregation method [11], the reference model assumes that nodes in two layers have highly correlated community labels. Specifically, we modified our probability model in Eq. (1) so that inter-layer interaction becomes a pairwise probability function $f(t_i(l), t_i(l'))$ instead of f_{check}. $f(t_i(l), t_i(l')) = 0.9$ if $t_i(l) = t_i(l')$, and 0.1 otherwise. We also derived Belief Propagation equations for the reference model. As the messages converge, the model encourages consistent labels for the same node between two layers.

We use a standard benchmark to test the performance of the algorithm, i.e the community detectability. In order to compare with single layer performance and a reference algorithm, we test our algorithm on a double-layer network, where each layer is generated from the same SBM, and we use the same SBM to generate one layer to test single layer performance. A test two-community network is generated as a single layer, where intra-community connectivity $p_{ab} = p_{in}$ if $a = b$, and inter-community connectivity $p_{ab} = p_{out}$ if $a \neq b$. Each community has 100 nodes. The ratio $\epsilon = p_{out}/p_{in}$ controls the quality of the generated community structure and is between 0 to 1. As ϵ increases, there is more association between two communities, and it is harder to distinguish them. A phase transition has been previously detected at a certain ϵ, where community detection accuracy drops significantly and communities become impossible to detect afterwards by any algorithm [19], [20]. We can characterize the detectability by the phase transition point.

The result is shown in Fig. 1. Using a double-layer setup, our algorithm, as well as the reference one, achieves significantly better detectability of the communities than single layer. That means some undetectable communities in single layers become detectable when considering both layers. Note that our proposed model does not a-priorily assume that the two layers are from the same SBM. Yet it still uncover which communities are consistent while they are not detectable individually. Thus our method has a comparable performance as the reference model.

For a more general multiplex network case, we compare our algorithm with a similar approach, GenLouvain [23], which is popular for multiplex network community detection and is also capable of identifying consistent communities across layers. GenLouvain applies a multilayer modularity maximization heuristics and has two main parameters, resolution parameter γ and interlayer coupling ω. When ω is set to 0, each layer

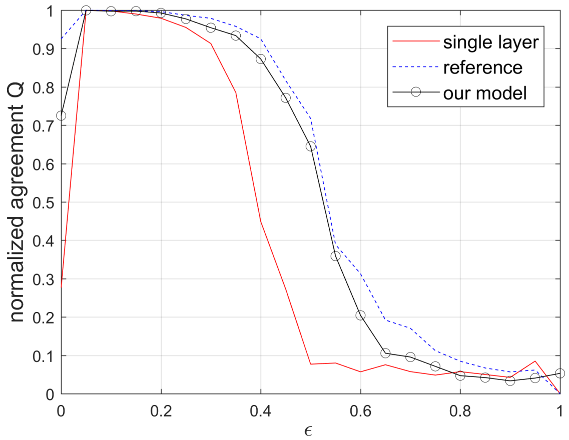

Fig. 1. Detectability transition curves of single layer network, multiplex network using our model, and a reference model. The reference model assumes two layers are from the same SBM and they have correlated community labels. Y axis shows overlap between detected community labels and the ground truth, given by normalized agreement [19]. The results are averaged over 20 experiments.

undergoes a community detection independently. When $\omega > 0$, same nodes in different layers may be assigned the same community labels. Since we assume no prior knowledge of consistent communities across layers, we set a uniform interlayer coupling parameter for all the nodes, which means all nodes have equal chance to have consistent community labels across layers. In contrast, our proposed algorithm similarly starts with homogeneous (random) interlayer coupling (messages). As messages update, our model effectively learns different weights for the interlayer coupling (messages), according to WPP constraints. Finally, for each node, our model will infer the probability of having consistent community labels across layers, in other words, strong couplings between consistent communities and weak couplings between inconsistent ones.

In a different experiment, we design a partially consistent two-layer network. The first layer has two communities (both comprised of 100 nodes) and the second one has three (100, 50 and 50 nodes), while one consistent community (100 nodes) exists in both layers. Therefore there are in total 4 different communities. The ideal output is that nodes in the consistent communities are assigned to the same community across layers, and the other inconsistent communities are also correctly detected themselves. If most (>50%) nodes are assigned labels in this ideal way, we consider the assignment correct. As a result, our model is able to converge to the correct point 39 times out of 100 trials with different initializations, while GenLouvain only achieves this 3 times. Note that GenLouvain may detect communities correctly in each layer if γ is chosen appropriately and interlayer coupling $\omega = 0$, but then the consistent communities will not be identified. On the contrary, if $\omega > 0$, GenLouvain has the problem of incorrectly assigning different communities the same label. The essential issue is that nodes should not have the same and constant interlayer coupling. This result demonstrates the problem we aim to solve here and the advantage of our model.

VI. Conclusion

In this paper we introduced a principled generative model for community structure in generic multiplex networks, with the constraint that a community label is associated with a consistent set of nodes if it appears in different layers. To keep the problem tractable, we also assume a node belongs to one community in each layer. Given the correct number of communities, the model is able to identify consistent communities across layers, and improve the community detectability. We derived message passing expressions using Belief Propagation to solve the probability model. Numerical experiments show our model achieves similar detectability improvement as a reference model which includes the prior knowledge that the layers are from the same SBM. This means that we need less information than knowing SBM consistency to yield the same performance. In contrast to previous studies which infer SBM parameters independently in each layer and then identify layer with the same SBM, our model unifies community detection within each layer and consistent community identification across layers into a single generative model. Moreover, rather than seeking layers with exactly the same SBM, our model allows two layers to have only part of their communities consistent and still remain identifiable. Finally, we applied our model to heterogeneous multiplex networks and compared with a popular method GenLouvain. The results indicate our model more reliably detects correct community labels. This work provides a novel idea for information fusion given mild prior knowledge. Since our algorithm detects block structures on adjacency matrices, the potential application and further study can be generalized to other fusion problems in the presence of multiple matrices.

References

[1] Ginestra Bianconi, "Statistical mechanics of multiplex networks: Entropy and overlap," *Physical Review E*, vol. 87, no. 6, pp. 062806, 2013.

[2] Manlio De Domenico, Albert Solé-Ribalta, Emanuele Cozzo, Mikko Kivelä, Yamir Moreno, Mason A. Porter, Sergio Gómez, and Alex Arenas, "Mathematical Formulation of Multilayer Networks," *Physical Review X*, vol. 3, no. 4, pp. 041022, 2013.

[3] Alessio Cardillo, Jesús Gómez-Gardeñes, Massimiliano Zanin, Miguel Romance, David Papo, Francisco del Pozo, and Stefano Boccaletti, "Emergence of network features from multiplexity," *Scientific Reports*, vol. 3, pp. 1344, 2013.

[4] S. Boccaletti, G. Bianconi, R. Criado, C. I. del Genio, J. Gómez-Gardeñes, M. Romance, I. Sendiña-Nadal, Z. Wang, and M. Zanin, "The structure and dynamics of multilayer networks," *Physics Reports*, vol. 544, no. 1, pp. 1, 2014.

[5] Manlio De Domenico, Vincenzo Nicosia, Alexandre Arenas, and Vito Latora, "Structural reducibility of multilayer networks," *Nature Communications*, vol. 6, pp. 6864, 2015.

[6] Shahin Mahdizadehaghdam, Han Wang, Hamid Krim, and Liyi Dai, "Information Diffusion of Topic Propagation in Social Media," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 2, no. 4, pp. 569, 2016.

[7] P. J. Mucha, Thomas Richardson, Kevin Macon, M. A. Porter, and J.-P. Onnela, "Community Structure in Time-Dependent, Multiscale, and Multiplex Networks," *Science*, vol. 328, no. 5980, pp. 876, 2010.

[8] Chuan Wen Loe and Henrik Jeldtoft Jensen, "Comparison of communities detection algorithms for multiplex," *Physica A: Statistical Mechanics and its Applications*, vol. 431, pp. 29–45, 2015.

[9] James D. Wilson, John Palowitch, Shankar Bhamidi, and Andrew B. Nobel, “Community Extraction in Multilayer Networks with Heterogeneous Community Structure,” arXiv:1610.06511, 2016.

[10] Toni Vallès-Català, Francesco A. Massucci, Roger Guimerà, and Marta Sales-Pardo, “Multilayer Stochastic Block Models Reveal the Multilayer Structure of Complex Networks,” *Physical Review X*, vol. 6, no. 1, pp. 011036, 2016.

[11] Dane Taylor, Saray Shai, Natalie Stanley, and Peter J. Mucha, “Enhanced Detectability of Community Structure in Multilayer Networks through Layer Aggregation,” *Physical Review Letters*, vol. 116, no. 22, pp. 228301, 2016.

[12] Natalie Stanley, Saray Shai, Dane Taylor, and Peter J. Mucha, “Clustering Network Layers with the Strata Multilayer Stochastic Block Model,” *IEEE Transactions on Network Science and Engineering*, vol. 3, no. 2, pp. 95, 2016.

[13] Nazanin Afsarmanesh and Matteo Magnani, “Finding overlapping communities in multiplex networks,” arXiv:1602.03746, 2016.

[14] Subhadeep Paul and Yuguo Chen, “Consistency of community detection in multi-layer networks using spectral and matrix factorization methods,” arXiv:1704.07353, 2017.

[15] Caterina De Bacco, Eleanor A. Power, Daniel B. Larremore, and Cristopher Moore, “Community detection, link prediction, and layer interdependence in multilayer networks,” *Physical Review E*, vol. 95, no. 4, pp. 042317, 2017.

[16] Qawi K. Telesford, Mary-Ellen Lynall, Jean Vettel, Michael B. Miller, Scott T. Grafton, and Danielle S. Bassett, “Detection of functional brain network reconfiguration during task-driven cognitive states,” *NeuroImage*, vol. 142, pp. 198, 2016.

[17] Michael Szell, Renaud Lambiotte, and Stefan Thurner, “Multirelational organization of large-scale social networks in an online world,” *Proceedings of the National Academy of Sciences*, vol. 107, no. 31, pp. 13636, 2010.

[18] Yuming Huang and Han Wang, “Consensus and multiplex approach for community detection in attributed networks,” in *2016 IEEE Global Conference on Signal and Information Processing (GlobalSIP),*. IEEE, pp. 425, 2016.

[19] Aurelien Decelle, Florent Krzakala, Cristopher Moore, and Lenka Zdeborová, “Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications,” *Physical Review E*, vol. 84, no. 6, pp. 066106, 2011.

[20] Aurelien Decelle, Florent Krzakala, Cristopher Moore, and Lenka Zdeborová, “Inference and phase transitions in the detection of modules in sparse networks,” *Physical Review Letters*, vol. 107, no. 6, pp. 065701, 2011.

[21] Yuchung J Wang and George Y Wong, “Stochastic Blockmodels for Directed Graphs,” *Journal of the American Statistical Association*, vol. 82, no. 397, pp. 8, 1987.

[22] Jonathan S Yedidia, William T Freeman, and Yair Weiss, “Understanding Belief Propagation and its Generalizations,” *Intelligence*, vol. 8, pp. 236, 2002.

[23] Lucas G. S. Jeub, Marya Bazzi, Inderjit S. Jutla, and Peter J. Mucha, “A generalized Louvain method for community detection implemented in MATLAB,” http://netwiki.amath.unc.edu/GenLouvain (2011-2017).