# Clustering of Diverse Multiplex Networks

Marianna Pensky and Yaxuan Wang

***Abstract*—The paper introduces the DIverse MultiPLEx Generalized Random Dot Product Graph (DIMPLE-GRDPG) network model where all layers of the network have the same collection of nodes and follow the Generalized Random Dot Product Graph (GRDPG) model. In addition, all layers can be partitioned into groups such that the layers in the same group are embedded in the same ambient subspace but otherwise all matrices of connection probabilities can be different. In a common particular case, where layers of the network follow the Stochastic Block Model, this setting implies that the groups of layers have common community structures but all matrices of block connection probabilities can be different. We refer to this version as the DIMPLE model. While the DIMPLE-GRDPG model generalizes the COmmon Subspace Independent Edge (COSIE) random graph model, the DIMPLE model includes a wide variety of SBM-equipped multilayer network models as its particular cases. In the paper, we introduce novel algorithms for the recovery of similar groups of layers, for the estimation of the ambient subspaces in the groups of layers in the DIMPLE-GRDPG setting, and for the within-layer clustering in the case of the DIMPLE model. We study the accuracy of those algorithms, both theoretically and via computer simulations. The advantages of the new models are demonstrated using simulations and real data examples.**

***Index Terms*—Community detection, multiplex network, spectral clustering, stochastic block model.**

## I. Introduction

### A. Multiplex Network Models

STOCHASTIC network models appear in a variety of applications, including genetics, proteomics, medical imaging, international relationships, brain science and many more. While in the early years of the field of stochastic networks, research mainly focused on studying a single network, in recent years the frontier moved to investigation of collection of networks, the so called *multilayer network*, which allows to study relationships between nodes with respect to various modalities (e.g., relationships between species based on food or space), or consists of network data collected from different individuals (e.g., brain networks). Although there are many different ways of modeling a multilayer network (see, e.g., an excellent review article of [1]), in this paper, we consider the case where all layers have the same set of nodes, and all the edges between nodes are drawn within layers, i.e., there are no edges connecting the nodes in different layers. Many authors, who work in a variety of research fields, study this particular version of a multilayer network (see, e.g., [2], [3], [4], [5], [6]). MacDonald et al. [6] called this type of multilayer network models the *Multiplex Network Model* and argued that it appears in a variety of real life situations.

Manuscript received 3 July 2023; revised 1 February 2024; accepted 28 February 2024. Date of publication 6 March 2024; date of current version 12 June 2024. This work was supported by the National Science Foundation (NSF) under Grant DMS-2014928 and Grant DMS-2310881. Recommended for acceptance by Prof. F. Radicchi. *(Corresponding author: Marianna Pensky.)*

The authors are with the Department of Mathematics, University of Central Florida, Orlando, FL 32816 USA (e-mail: marianna.pensky@ucf.edu).

This article has supplementary downloadable material available at https://doi.org/10.1109/TNSE.2024.3374102, provided by the authors.

Digital Object Identifier 10.1109/TNSE.2024.3374102

### B. Popular Stochastic Network Models

In this paper, we shall consider two types of network models. The most popular Stochastic Block Model (**SBM**), introduced in [7], assumes that all nodes of the network can be divided into the communities, and that the connection probability between a pair of nodes is fully determined by the communities to which those nodes belong. Denote $[n] = \{1, \ldots, n\}$. If $n$ nodes are divided into $K$ communities, then there exists a community assignment function $z : [n] \to [K]$, and the block probability matrix $\mathbf{B}$, such that the probability of connection between nodes $i$ and $j$ is $\mathbf{P}(i, j) = \mathbf{B}(z(i), z(j))$. Alternatively, one can introduce a clustering matrix $\mathbf{Z} \in \{0, 1\}^{n\times K}$ such that $\mathbf{Z}(i, k) = 1$ if node $i$ belongs to community $k$ and $\mathbf{Z}(i, k) = 0$ otherwise, and set $\mathbf{P} = \mathbf{Z}\mathbf{B}\mathbf{Z}^T$.

The popularity of the SBM is due to the fact that the SBM, according to [8], provides a universal tool for description of time-independent stochastic network data. It is also very common in applications. For example, [9] argues that stochastic block models provide a powerful tool for brain studies (see also [10], [11], [12]).

Nevertheless, the SBM fails to describe many real life networks that exhibit high degree of heterogeneity between the nodes. For this reason, in the last two decades a variety of more flexible models were introduced. All those models, however, can be viewed as particular cases of the so called Generalized Random Dot Product Graph (**GRDPG**) model [13] which assumes that the matrix of connection probabilities $\mathbf{P}$ can be presented as $\mathbf{P} = \mathbf{X}\mathbf{I}_{p,q}\mathbf{X}^T$ where $\mathbf{X} \in \mathbb{R}^{n\times K}$ is the latent position matrix and $\mathbf{I}_{p.q}$ is the diagonal matrix with $p$ ones and $q$ negative ones on the diagonal, $p + q = K$. Matrix $\mathbf{X}$ is assumed to be such that $\mathbf{P} \in [0, 1]^{n\times n}$. If $\mathbf{X} = \mathbf{V}\mathbf{D}_X\mathbf{V}_X$ is the Singular Value Decomposition (**SVD**) of $\mathbf{X}$, then $\mathbf{P}$ can be alternatively presented as $\mathbf{P} = \mathbf{V}\mathbf{Q}\mathbf{V}^T$, where $\mathbf{Q} = \mathbf{D}_X\mathbf{V}_X\mathbf{I}_{p,q}(\mathbf{V}_X)^T\mathbf{D}_X$. Here, $\mathbf{V}$ is the basis of the ambient subspace of the GRDPG model and $\mathbf{Q}$ is the loading matrix. Since in the SBM, matrix $\mathbf{V} = \mathbf{Z}((\mathbf{Z})^T\mathbf{Z})^{-1/2}$ has orthonormal columns, it is evident that the SBM is a particular case of GRDPG with $\mathbf{Q} = ((\mathbf{Z})^T\mathbf{Z})^{1/2}\mathbf{B}((\mathbf{Z})^T\mathbf{Z})^{1/2}$ and $\mathbf{V}$ given above.

In what follows we shall consider a multiplex network with layers that follow either GRDPG or SBM In addition, we shall allow diversity between the layers.

### C. DIverse MultiPLEx (DIMPLE) Network Models Frameworks

Consider an $L$-layer network on the same set of $n$ vertices, where the tensor of probabilities of connections $\mathcal{P} \in [0,1]^{n\times n\times L}$ is formed by layers $\mathbf{P}^{(l)}$, $l \in [L]$, that can be partitioned into $M$ groups with the common subspace structure or community assignment. The latter means that there exists a label function $c : [L] \to [M]$ which identifies to which of $M$ groups a layer belongs.

If the layers of the network follow the GRDPG model, we assume that each group of $M$ layers is embedded in its own ambient subspace but all loading matrices can be different. Specifically, $\mathbf{P}^{(l)}$, $l \in [L]$, are given by

$$\mathbf{P}^{(l)} = \mathbf{V}^{(m)}\mathbf{Q}^{(l)}\left(\mathbf{V}^{(m)}\right)^T, \quad m = c(l),\ m \in [M], \qquad (1)$$

where $\mathbf{Q}^{(l)} = (\mathbf{Q}^{(l)})^T$ and $\mathbf{V}^{(m)} \in \mathbb{R}^{n\times K_m}$ is a basis matrix of the $m$-th group of layers' subspace, $(\mathbf{V}^{(m)})^T\mathbf{V}^{(m)} = \mathbf{I}$, such that all entries of $\mathbf{P}^{(l)}$ are in [0,1]. We shall call this model the DIverse MultiPLEx Generalized Random Dot Product Graph (**DIMPLE-GRDPG**).

In the case, when layers of the network follow the SBM, the groups of layers have common community structures but matrices of block connection probabilities can be all different. Then,

$$\mathbf{P}^{(l)} = \mathbf{Z}^{(m)}\mathbf{B}^{(l)}\left(\mathbf{Z}^{(m)}\right)^T, \quad m = c(l),\ m \in [M], \qquad (2)$$

where $\mathbf{Z}^{(m)}$ is the clustering matrix in the layer of type $m = c(l)$, that partitions $n$ nodes into $K_m$ communities, and $\mathbf{B}^{(l)} = (\mathbf{B}^{(l)})^T$ is a matrix of block probabilities, $l \in [L]$. In order to distinguish this special case, we shall refer to (2) as simply the **DIMPLE** model.

In both models, one observes the adjacency tensor $\mathcal{A} \in \{0,1\}^{n\times n\times L}$ with layers $\mathbf{A}^{(l)}$ such that $\mathbf{A}^{(l)}(i,j) = \mathbf{A}^{(l)}(j,i)$ and, for $1 \le i < j \le n$ and $l \in [L]$, where $\mathbf{A}^{(l)}(i,j)$ are the Bernoulli random variables with $\mathbb{P}(\mathbf{A}^{(l)}(i,j) = 1) = \mathbf{P}^{(l)}(i,j)$, and they are independent from each other. The objective is to recover the layer clustering function $c : [L] \to [M]$, and the community assignment matrices $\mathbf{Z}^{(m)}$ or subspace bases matrices $\mathbf{V}^{(m)}$ in the case of models (2) or (1), respectively.

Note that, since the SBM is a particular case of the GRDPG, (2) is a particular case of (1). Nevertheless, the problems associated with (1) and (2) are somewhat different. While recovering matrices $\mathbf{V}^{(m)}$ is an estimation problem, finding communities in the groups of layers, corresponding to clustering matrices $\mathbf{Z}^{(m)}$, is a clustering problem. For this reason, we study both models, (1) and (2), in this paper.

Our paper makes several key contributions.

1) To the best of our knowledge, our paper studies the most general multiplex network model which has been examined in the statistical network literature so far. In particular, we allow the layers of the network to be equipped with the most flexible GRDPG model where layers have $M$ versions of the subspace structures and all loading matrices are different.
2) For this reason, our paper generalizes a multitude of publications that investigate multiplex stochastic networks. We describe the types of models that DIMPLE-GRDPG generalizes in the next section. The advantage of our approach is that the methodologies of this paper can be successfully applied to all those particular cases without making a particular choice of a model, but the reverse is not true. We confirm this via simulations.
3) Our paper develops a novel between-layer clustering algorithm that works for both DIMPLE and DIMPLE-GRDPG network model and derive expressions for the clustering errors under very simple and intuitive assumptions. Our simulations confirm that the between-layer and the within-layer clustering algorithms deliver high precision in a finite parameter settings. In addition, if $M = 1$, our subspace recovery error compares favorably to the ones in [14] and [15], due to employment of a different algorithm.
4) Since the DIMPLE-GRDPG and the DIMPLE network models generalize a multitude of more restrictive models, our paper opens a gateway for testing/model selection. In particular, one can test whether subspaces /communities persist throughout the layers of the network, or whether layers should be partitioned into several groups, which is equivalent to testing the hypothesis that $M = 1$ in (1) or (2).

### D. Justification of the Model and Related Work

In the last few years, a number of authors studied multiplex network models. The vast majority of the paper assumed that all layers of the network follow the SBM.

While the scientific community considered various types of multiplex networks in general, and the SBM-equipped multiplex networks in particular (see e.g., [16], [17] among others), the theoretically inclined papers in the field of statistics mainly have been investigating the case where communities persist throughout all layers of the network while the matrices of block connection probabilities can take arbitrary values (see, e.g., [18], [19], [20], [21] and references therein). This case corresponds to $M = 1$ in (2).

In the GRDPG-equipped networks, the COmmon Subspace Independent Edge (**COSIE**) random graph model of [14] and [15] follows a similar approach and assumes that every layer of the network is embedded into the same ambient subspace, which corresponds to $M = 1$ in (1).

Nevertheless, there are many real life scenarios where the assumption, that all layers of the network have the same communities or are embedded into the same subspace is too restrictive. For example, it is known that some brain disorders are associated with changes in brain network organizations (see, e.g., [22]).

One of the possible approaches here is to assume that both, the community structures and the probabilities of connections in the network layers, will be identical under the same biological condition and dissimilar for different conditions. This type of setting, called the **M**ixture **M**ulti**L**ayer **S**tochastic **B**lock **M**odel (**MMLSBM**) assumes that all layers can be partitioned into a few different types,such that each distinct type of layers is equipped

with its own community structure and a unique matrix of block connection probabilities, and that both are identical within the same type of layers.

Specifically, if $M = 1$, then the DIMPLE model (2) reduces to the multiplex models in [18], [19], [20], [21], [23] with the persistent communities, and it becomes the MMLSBM of [24], [25] and [26], if $\mathbf{B}^{(l)}$ takes only $M$ distinct values, i.e., $\mathbf{B}^{(l)} = \mathbf{B}^{(m)}$ for $c(l) = m$. Similarly, if $M = 1$, the DIMPLE-GRDPG model in (1) reduces to the COSIE model in [14] and [15]. and it reduces to a low rank tensor estimation of [27] and [28] if all matrices $\mathbf{Q}^{(l)}$ are identical within a group of layers.

Hence, so far authors considered two complementary types of settings for multiplex networks. In the first of them, all layers of the network are embedded into the same subspaces in the case of the GRDPG, or have the same communities if the layers of the network are equipped with SBMs. In the second one, the layers may be embedded into different subspaces, but the tensor of connection probabilities has a low rank, which reduces to MMLSBM if layers follow the SBM.

Therefore, the natural generalization of those two scenarios would be the setting, where the layers of the network can be partitioned into groups, each with the distinct subspace or community structure. Such multiplex network can be viewed as a concatenation of several multiplex networks that follow COSIE model or Stochastic Block Models with persistent community structure. On the other hand, such networks will reduce to a low rank tensor or the MMLSBM if networks in the group of layers have identical probabilities of connections.

The real data examples in Section V show advantages of the setting where layers of the network are partitioned into groups rather than being embedded in the same subspace while simultaneously demonstrate that the assumption of the MMLSBM, that each of these groups of layers have identical matrices of connection probabilities is equally problematic.

The new DIMPLE-GRDPG model requires development of new algorithms, since the probability tensor $\mathcal{P}$ associated with the DIMPLE-GRDPG model in (1) does not have a low rank, due to the fact that all matrices $\mathbf{Q}^{(l)}$ are different. For this reason, techniques and theoretical assessments developed for low rank tensors do not work in the case of the DIMPLE-GRDPG model. Similarly, since the matrices of the block connection probabilities take different values in each of the layers, techniques employed in [25] and [26] cannot be applied in the new environment of DIMPLE.

In Section IV we show that while our methodology works well for the MMLSBM (although, as it is expected, it is less efficient for small values of $n$ and $L$ since it cannot take advantage of the more restricted structure of the MMLSBM), the MMLSBM-based algorithms fail in the case of our, more flexible model.

Indeed, the TWIST algorithm of [25] relies on the fact that the tensor of connection probabilities is truly low rank in the case of MMLSBM. This, however, is not true for the DIMPLE model, where the matrices of block connection probabilities vary from layer to layer. On the other hand, the ALMA algorithm of [26] exploits the fact that the matrices of connection probabilities are identical in the groups of layers with the same community structures. This is no longer true in the environment of the DIMPLE model, where matrices of connection probabilities are all different for different layers.

### E. Notations

For any integer $n$, we denote $[n] = \{1, \ldots, n\}$. We denote tensors by calligraphy letters and matrices by bold letters. Denote by $\mathcal{M}_{N,K}$ the set of the *clustering* matrices for $N$ objects partitioned into $K$ groups

$$\mathcal{M}_{N,K} = \left\{\mathbf{X} \in \{0,1\}^{N\times K}, \quad \mathbf{X}\mathbf{1} = \mathbf{1}, \quad \mathbf{X}^T\mathbf{1} \neq \mathbf{0}\right\},$$

where $\mathbf{X} \in \mathcal{M}_{N,K}$ are such that $\mathbf{X}_{i,j} = 1$ if node $i$ is in cluster $j$ and and $\mathbf{X}_{i,j} = 0$ otherwise. For any matrix $\mathbf{X}$, denote the Frobenius, the infinity and the operator norm by $\|\mathbf{X}\|_F$, $\|\mathbf{X}\|_\infty$ and $\|\boldsymbol{X}\|$, respectively, and its $r$-th largest singular value by $\sigma_r(\mathbf{X})$. The column $j$ and the row $i$ of a matrix $\mathbf{Q}$ are denoted by $\mathbf{Q}(:, j)$ and $\mathbf{Q}(i, :)$, respectively. Denote the identity and the zero matrix of size $K$ by, respectively, $\mathbf{I}_K$ and $\mathbf{0}_K$ (where $K$ is omitted when this does not cause ambiguity). Denote

$$\mathcal{O}_{n,K} = \left\{\mathbf{X} \in \mathbb{R}^{n\times K} : \ \mathbf{X}^T\mathbf{X} = \mathbf{I}_K\right\}, \quad \mathcal{O}_n = \mathcal{O}_{n,n}. \quad (3)$$

Let $\mathrm{vec}(\mathbf{X})$ be the vector obtained from matrix $\mathbf{X}$ by sequentially stacking its columns. Denote by $\mathbf{X} \otimes \mathbf{Y}$ the Kronecker product of matrices $\mathbf{X}$ and $\mathbf{Y}$. Denote $n$-dimensional vector with unit components by $\mathbf{1}_n$. Denote diagonal of a matrix $\mathbf{A}$ by $\mathrm{diag}(\mathbf{A})$. Also, denote the $M$-dimensional diagonal matrix with $a_1, \ldots, a_M$ on the diagonal by $\mathrm{diag}(a_1, \ldots, a_M)$.

For any matrix $\mathbf{X} \in \mathbb{R}^{n_1\times n_2}$, denote its projection on the nearest rank $K$ matrix by $\Pi_K(\mathbf{X})$. For any matrices $\mathbf{X} \in \mathbb{R}^{n_1\times n_2}$ and $\mathbf{U} \in \mathbf{O}_{n_1,K}$, $K \leq n_1$, projection of $\mathbf{X}$ on the column space of $\mathbf{U}$ and on its orthogonal space are defined, respectively, as $\Pi_{\mathbf{U}}(\mathbf{X}) = \mathbf{U}\mathbf{U}^T\mathbf{X}$, $\Pi_{\mathbf{U}_\perp}(\mathbf{X}) = (\mathbf{I} - \Pi_{\mathbf{U}})\mathbf{X}$. Following [29], we define the following tensor operations. For any tensor $\mathcal{X} \in \mathbb{R}^{n_1\times n_2\times n_3}$ and a matrix $\mathbf{A} \in \mathbb{R}^{m\times n_3}$, their product $\mathcal{X} \times_3 \mathbf{A}$ along dimension 3 is a tensor in $\mathbb{R}^{n_1\times n_2\times m}$ with elements

$$[\mathcal{X} \times_3 \mathbf{A}](i_1, i_2, j) = \sum_{i_3=1}^{n_3} \mathbf{A}(j, i_3)\mathcal{X}(i_1, i_2, i_3), \quad j \in [m].$$

If $\mathcal{Y} \in \mathbb{R}^{m\times n_2\times n_3}$ is another tensor, the product between tensors $\mathcal{X}$ and $\mathcal{Y}$ along dimensions (2,3), denoted by $\mathcal{X} \times_{2,3} \mathcal{Y}$, is a matrix in $\mathbb{R}^{n_1\times m}$ with elements

$$[\mathcal{X} \times_{2,3} \mathcal{Y}](i_1, i_2) = \sum_{j_2=1}^{n_2}\sum_{j_3=1}^{n_3} \mathcal{X}(i_1, j_2, j_3)\mathcal{Y}(i_2, j_2, j_3),$$

for $i_1 \in [n_1]$, $i_2 \in [m]$. The mode-3 matricization of tensor $\mathcal{X} \in \mathbb{R}^{n_1\times n_2\times n_3}$ is a matrix $\mathscr{M}_3(\mathcal{X}) = \mathbf{X} \in \mathbb{R}^{n_3\times(n_1n_2)}$ with rows $\mathbf{X}(i, :) = [\mathrm{vec}(\mathcal{X}(:, :, i))]^T$. Please, see [29] for a more extensive discussion of tensor operations and their properties.

We use the $\sin\Theta$ distances to measure the separation between two subspaces with orthonormal bases $\mathbf{U} \in \mathcal{O}_{n,K}$ and $\widetilde{\mathbf{U}} \in \mathcal{O}_{n,K}$, respectively. Suppose the singular values of $\mathbf{U}^T\widetilde{\mathbf{U}}$ are $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_K >> 0$. Then $\Theta(\mathbf{U}, \widetilde{\mathbf{U}}) = \mathrm{diag}(\cos^{-1}(\sigma_1), \ldots, \cos^{-1}(\sigma_K))$ are the principle angles. Quantitative measures of the distance between the column spaces of

$\mathbf{U}$ and $\widetilde{\mathbf{U}}$ are then

$$\left\|\sin\Theta\left(\mathbf{U},\widetilde{\mathbf{U}}\right)\right\| = \sqrt{1-\sigma^2_{\min}\left(\mathbf{U}^T\widetilde{\mathbf{U}}\right)},$$
$$\left\|\sin\Theta\left(\mathbf{U},\widetilde{\mathbf{U}}\right)\right\|_F = \sqrt{K-\|\mathbf{U}^T\widetilde{\mathbf{U}}\right)\|^2_F}. \quad (4)$$

Some convenient characterizations of those distances can be found in Section 8.1 of [30].

Finally, we shall use $C$ for a generic positive constant that can take different values and is independent of $L$, $n$, $M$, $K$ and graph densities.

## II. Fitting the DIMPLE and the DIMPLE-GRDPG Models

In this paper, we consider a multiplex network with $L$ layers of $M$ types, where $L_m$ is the number of layers of type $m \in [M]$. Let $\mathbf{C} \in \mathcal{M}(L,M)$ be the layer clustering matrix. A layer of type $m$ has an ambient dimension $K_m$. In the case of model (2), a layer of type $m$ has $K_m$ communities, and $n_{k,m}$ is the number of nodes of type $k \in [K_m]$ in the layer of type $m$, so that

$$\mathbf{D}_z^{(m)} = \left(\mathbf{Z}^{(m)}\right)^T \mathbf{Z}^{(m)} = \text{diag}(n_{1,m},\ldots,n_{K_m,m}). \quad (5)$$

### A. Between-Layer Clustering

First, we note that model (2) is a particular case of model (1). Indeed, denote $\mathbf{U}_z^{(m)} = \mathbf{Z}^{(m)}(\mathbf{D}_z^{(m)})^{-1/2}$, where matrices $\mathbf{D}_z^{(m)}$ are defined in (5). Since $\mathbf{U}_z^{(m)} \in \mathcal{O}_{n,K_m}$, matrices $\mathbf{P}^{(l)}$ in (2) can be written as

$$\mathbf{P}^{(l)} = \mathbf{U}_z^{(m)}\mathbf{B}_D^{(l)}\left(\mathbf{U}_z^{(m)}\right)^T,\ \mathbf{B}_D^{(l)} = \sqrt{\mathbf{D}_z^{(m)}}\mathbf{B}^{(l)}\sqrt{\mathbf{D}_z^{(m)}}. \quad (6)$$

Therefore, (2) is a particular case of (1) with $\mathbf{V}^{(m)} = \mathbf{U}_z^{(m)}$ and $\mathbf{Q}^{(l)} = \mathbf{B}_D^{(l)}$. For this reason, we are going to cluster groups of layers in the more general setting (1) of DIMPLE-GRDPG.

In order to find the clustering matrix $\mathbf{C}$, observe that matrices $\mathbf{P}^{(l)}$ in (1) can be written as

$$\mathbf{P}^{(l)} = \mathbf{V}^{(m)}\mathbf{O}_Q^{(l)}\mathbf{S}_Q^{(l)}\left(\mathbf{O}_Q^{(l)}\right)^T\left(\mathbf{V}^{(m)}\right)^T,\quad l\in[L], \quad (7)$$

where

$$\mathbf{Q}^{(l)} = \mathbf{O}_Q^{(l)}\mathbf{S}_Q^{(l)}\left(\mathbf{O}_Q^{(l)}\right)^T,\quad l\in[L], \quad (8)$$

is the singular value decomposition (**SVD**) of $\mathbf{Q}^{(l)}$ with $\mathbf{O}_Q^{(l)} \in \mathcal{O}_{n,K_m}$, $m = c(l)$, and diagonal matrix $\mathbf{S}_Q^{(l)}$. In order to extract common information from matrices $\mathbf{P}^{(l)}$, we consider the SVD of $\mathbf{P}^{(l)}$, $l \in [L]$,

$$\mathbf{P}^{(l)} = \mathbf{U}_{P,l}\Lambda_{P,l}(\mathbf{U}_{P,l})^T,\quad \mathbf{U}_{P,l}\in\mathcal{O}_{n,K_m},\ m=c(l), \quad (9)$$

and relate it to the expansion (7). If, as we assume later, matrices $\mathbf{Q}^{(l)}$ are of full rank, then $\mathbf{O}_Q^{(l)} \in \mathcal{O}_{K_m}$, so that $\mathbf{O}_Q^{(l)}\left(\mathbf{O}_Q^{(l)}\right)^T = \left(\mathbf{O}_Q^{(l)}\right)^T\mathbf{O}_Q^{(l)} = \mathbf{I}_{K_m}$, $m = c(l)$. Therefore, $\mathbf{V}^{(m)}\mathbf{O}_Q^{(l)} \in \mathcal{O}_{n,K_m}$, and expansion (7) is just another way of writing the SVD of $\mathbf{P}^{(l)}$. Hence, when $c(l) = m$, one has $\mathbf{U}_{P,l} = \mathbf{V}^{(m)}\mathbf{O}_V^{(l)}$ where $\mathbf{O}_V^{(l)} \in \mathcal{O}_{K_m}$ is a $K_m$-dimensional rotation. Since matrices $\mathbf{O}_V^{(l)}$ are unknown, we introduce alternatives to $\mathbf{U}_{P,l}$:

$$\mathbf{U}_{P,l}\left(\mathbf{U}_{P,l}\right)^T = \mathbf{V}^{(m)}\left(\mathbf{V}^{(m)}\right)^T,\quad m=c(l), \quad (10)$$

which depend on $l$ only via $m = c(l)$ and are uniquely defined for $l \in [L]$. The latter implies that the between-layer clustering can be based on the matrices $\mathbf{U}_{P,l}(\mathbf{U}_{P,l})^T$, $l \in [L]$, or rather on their vectorized versions. Denote

$$\mathbf{D}_c = \mathbf{C}^T\mathbf{C} = \text{diag}(L_1,\ldots,L_M),\ \mathbf{W} = \mathbf{C}(\mathbf{D}_c)^{-1/2}, \quad (11)$$

where $\mathbf{W} \in \mathcal{O}_{L,M}$.

Consider matrices $\Theta \in \mathbb{R}^{n^2\times L}$ and $\Psi \in \mathbb{R}^{n^2\times M}$ with respective columns $\Theta(:,l) = \text{vec}(\mathbf{V}^{(c(l))}(\mathbf{V}^{(c(l))})^T) = \text{vec}(\mathbf{U}_{P,l}(\mathbf{U}_{P,l})^T)$ and $\Psi(:,m) = \text{vec}(\mathbf{V}^{(m)}(\mathbf{V}^{(m)})^T)$, where $m \in [M]$, $l \in [L]$. Then,

$$\Theta = \Psi\mathbf{C}^T,\quad \Psi = \Theta\mathbf{C}\mathbf{D}_c^{-1}, \quad (12)$$

so that clustering assignment can be recovered by spectral clustering of columns of an estimated version of matrix $\Theta$.

For this purpose, consider layers $\mathbf{A}^{(l)} = \mathcal{A}(:,:,l)$ of the adjacency tensor $\mathcal{A}$ and construct the SVDs of their rank $K_m$ projections $\Pi_{K_m}(\mathbf{A}^{(l)})$, $m = c(l)$, $l \in [L]$:

$$\Pi_{K_m}\left(\mathbf{A}^{(l)}\right) = \widehat{\mathbf{U}}_{A,l}\widehat{\Lambda}_{P,l}\left(\widehat{\mathbf{U}}_{A,l}\right)^T,\ \widehat{\mathbf{U}}_{A,l}\in\mathcal{O}_{n,K_m} \quad (13)$$

Then, replace matrix $\Theta$ by its proxy $\widehat{\Theta}$ with columns $\widehat{\Theta}(:,l) = \text{vec}(\widehat{\mathbf{U}}_{A,l}(\widehat{\mathbf{U}}_{A,l})^T)$. The major difference between $\Theta$ and $\widehat{\Theta}$, however, is that, under assumptions in Section III-A, $\text{rank}(\Theta) = M$ while, in general, $\text{rank}(\widehat{\Theta}) = L \gg M$. If the SVD of $\widehat{\Theta}$ is

$$\widehat{\Theta} = \widetilde{\mathcal{V}}\widetilde{\Lambda}\widetilde{\mathcal{W}},\quad \widetilde{\mathcal{V}}\in\mathcal{O}_{n^2,L},\quad \widetilde{\mathcal{W}}\in\mathcal{O}_L, \quad (14)$$

then, we can form reduced matrices

$$\widehat{\mathcal{V}} = \widetilde{\mathcal{V}}(:,1:M)\in\mathcal{O}_{n^2,M},\ \widehat{\mathcal{W}} = \widetilde{\mathcal{W}}(:,1:M)\in\mathcal{O}_{L,M},$$

---

**Algorithm 1:** The Between-Layer Clustering.

**Input:** Adjacency tensor $\mathcal{A} \in \{0,1\}^{n\times n\times L}$; number of groups of layers $M$; ambient dimension $K^{(l)}$ of each layer $l \in [L]$; parameter $\epsilon$

**Output:** Estimated clustering matrix $\widehat{\mathbf{C}} \in \mathcal{M}_{L,M}$

**Steps:**

**1:** Find the SVDs $\Pi_{K^{(l)}}(\mathbf{A}^{(l)}) = \widehat{\mathbf{U}}_{A,l}\widehat{\Lambda}_{P,l}(\widehat{\mathbf{U}}_{A,l})^T$, $\widehat{\mathbf{U}}_{A,l}\in\mathcal{O}_{n^2,K^{(l)}}$, $l\in[L]$

**2:** Form matrix $\widehat{\Theta}\in\mathbb{R}^{n^2\times L}$ with columns $\widehat{\Theta}(:,l) = \text{vec}\left(\widehat{\mathbf{U}}_{A,l}(\widehat{\mathbf{U}}_{A,l})^T\right)$

**3:** Construct the SVD of $\widehat{\Theta}$ using (14) and obtain matrix $\widehat{\mathcal{W}} = \widetilde{\mathcal{W}}(:,1:M)\in\mathcal{O}_{L,M}$

**4:** Cluster $L$ rows of $\widehat{\mathcal{W}}$ into $M$ clusters using $(1+\epsilon)$-approximate $K$-means clustering. Obtain estimated clustering matrix $\widehat{\mathbf{C}}$

---

and apply clustering to the rows of $\widehat{\mathcal{W}}$ rather than to the rows of $\widetilde{\mathcal{W}}$. The latter results in Algorithm 1. We use $(1+\epsilon)$-approximate $K$-means clustering to obtain the final clustering assignments. There exist efficient algorithms for solving the $(1+\epsilon)-$approximate $K$-means problem (see, e.g., [31]). We denote

$$\widehat{\mathbf{D}}_c = \widehat{\mathbf{C}}^T\widehat{\mathbf{C}}, \quad \widehat{\mathbf{W}} = \widehat{\mathbf{C}}\widehat{\mathbf{D}}_c^{-1/2} \in \mathcal{O}_{L,M}. \tag{15}$$

Observe that clustering procedure above relies on the knowledge of the ambient dimension $K_m$, which is associated with the unknown group membership $m = c(l)$. Instead of assuming that $K_m$ are known, as it is done in [25] and [26], we assume that one knows the ambient dimension $K^{(l)}$ of the GRDPG in every layer $l \in [L]$ of the network. This is a very common assumption and is imposed in almost every paper that studies latent position or block model equipped networks (see, e.g., [13], [32], [33], [34]). In this case, one can replace $K_m$ in (13) by $K^{(l)}$. We further discuss this issue in Remark 2.

*Remark 1: Unknown number of layers.* While Algorithm 1 assumes $M$ to be known, in many practical situations this is not true, and the value of $M$ has to be discovered from data. Identifying the number of clusters is a common issue in data clustering, and it is a separate problem from the process of actually solving the clustering problem with a known number of clusters. There are many ways for estimating the number of clusters, for instance, by evaluation of the clustering error in terms of an objective function, as in, e.g., [35], or by monitoring the eigenvalues of the non-backtracking matrix or the Bethe Hessian matrix, as it is done in [36].

*Remark 2: Unknown ambient dimensions.* In this paper, for the purpose of methodological developments, we assume that the ambient dimension (number of communities in the case of the DIMPLE model) $K^{(l)}$ of each layer of the network is known. This is a common assumption, and everything in the Remark 1 can also be applied to this case. Here, $K^{(l)} = K_m$ with $m = c(l)$. One can, of course, can assume that the values of $K_m, m \in [M]$, are known. However, since group labels are interchangeable, in the case of non-identical subspace dimensions (numbers of communities), it is hard to choose, which of the values corresponds to which of the groups. This is actually the reason why [25] and [26], who imposed this assumption, used it only in theory, while their simulations and real data examples are all restricted to the case of equal number of communities in all layers $K_m = K, m \in [M]$. On the contrary, knowledge of $K^{(l)}$ allows one to deal with different ambient dimensions (number of communities) in the groups of layers in simulations and real data examples. Specifically, one can infer the ambient dimension in each group of layers using the ScreeNOT technique of [37].

### *B. Fitting Invariant Subspaces in Groups of Layers in the DIMPLE-GRDPG Model. Within-Layer Clustering in the DIMPLE Model*

If we knew the true clustering matrix $\mathbf{C}$ and the true probability tensor $\mathcal{P} \in \mathbb{R}^{n\times n\times L}$ with layers $\mathbf{P}^{(l)}$ given by (1), then we could average layers with identical subspace structures. In the case of real data, however, precision of estimating $\mathbf{V}^{(m)}$ depends on the lowest nonzero eigenvalue of the sum of of $\mathbf{Q}^{(l)}$ with $c(l) = m$. Since eigenvalues of $\mathbf{Q}^{(l)}$ can be negative and positive, the lower bound on the latter is not guaranteed. Alternatively, one can add the squares $\mathbf{G}^{(l)} = (\mathbf{P}^{(l)})^2$, obtaining, for $m \in [M]$,

$$\sum_{c(l)=m} \mathbf{G}^{(l)} = \sum_{c(l)=m} \left(\mathbf{P}^{(l)}\right)^2 = \sum_{c(l)=m} \mathbf{V}^{(m)}\left(\mathbf{Q}^{(l)}\right)^2\left(\mathbf{V}^{(m)}\right)^T.$$

In this case, the eigenvalues of $(\mathbf{Q}^{(l)})^2$ are all positive which ensures successful recovery of matrices $\mathbf{V}^{(m)}$.

Note that, however, $(\mathbf{A}^{(l)})^2$ is not an unbiased estimator of $(\mathbf{P}^{(l)})^2$. Indeed, while $\mathbb{E}((\mathbf{A}^{(l)})^2)_{i,j} = ((\mathbf{P}^{(l)})^2)_{i,j}$ for $i \neq j$, for the diagonal elements, one has

$\mathbb{E}((\mathbf{A}^{(l)})^2)_{i,i} = (\mathbf{P}^{(l)})^2_{i,i} + \sum_j[(\mathbf{P}^{(l)})_{i,j} - (\mathbf{P}^{(l)})^2_{i,j}]$.

Therefore, following [19], we evaluate the degree vector $\hat{\mathbf{d}}^{(l)} = \mathbf{A}^{(l)}\mathbf{1}_n$ and form diagonal matrices $\mathrm{diag}(\hat{\mathbf{d}}^{(l)})$ with vectors $\hat{\mathbf{d}}^{(l)}$ on the diagonals. We construct a tensor $\widehat{\mathcal{G}} \in \mathbb{R}^{n\times n\times L}$ with layers $\widehat{\mathbf{G}}^{(l)} = \widehat{\mathcal{G}}(:,:,l)$ of the form

$$\widehat{\mathbf{G}}^{(l)} = \left(\mathbf{A}^{(l)}\right)^2 - \mathrm{diag}(\hat{\mathbf{d}}^{(l)}), \quad l \in [L]. \tag{16}$$

Subsequently, we combine layers of the same types, obtaining tensor $\widehat{\mathcal{H}} \in \mathbb{R}^{n\times n\times M}$

$$\widehat{\mathcal{H}} = \widehat{\mathcal{G}} \times_3 \widehat{\mathbf{W}}^T, \tag{17}$$

where $\widehat{\mathbf{W}}$ is defined in (15). After that, $\mathbf{V}^{(m)}$, $m \in [M]$, can be estimated using the SVD. The procedure is described in Algorithm 2.

After the matrices $\mathbf{V}^{(m)}$ have been estimated, in the case of the DIMPLE model, one can find the clustering matrices $\mathbf{Z}^{(m)}$ in (2) by approximate $K$-means clustering. Indeed, up to a rotation, $\mathbf{V}^{(m)}$ is equal to $\mathbf{U}_z^{(m)} = \mathbf{Z}^{(m)}(\mathbf{D}_z^{(m)})^{-1/2}$, where $\mathbf{Z}^{(m)}$ is the

**Algorithm 2:** Estimating Invariant Subspaces (DIMPLE-GRDPG Model) and Within-Layer Clustering (DIMPLE Model).

**Input:** Adjacency tensor $\mathcal{A} \in \{0,1\}^{n\times n\times L}$; number of groups of layers $M$; ambient dimensions $K_m$, $m \in [M]$, of each group of layers; estimated clustering matrix $\widehat{\mathbf{C}} \in \mathcal{M}_{L,M}$

**Output:** For $m \in [M]$, estimated invariant subspaces $\widehat{\mathbf{V}}^{(m)}$, (DIMPLE-GRDPG model); estimated clustering matrices $\widehat{\mathbf{Z}}^{(m)}$ (DIMPLE model)

**Steps:**

**1:** Construct tensor $\widehat{\mathcal{G}}$ with layers $\widehat{\mathbf{G}}^{(l)}$ given by (16), $l \in [L]$

**2:** Construct tensor $\widehat{\mathcal{H}}$ using formula (17)

**3:** Construct the SVDs of layers $\widehat{\mathbf{H}}^{(m)} = \widetilde{\mathbf{U}}_{\widehat{\mathbf{H}}}^{(m)}\widehat{\Lambda}_{\widehat{\mathbf{H}}}^{(m)}(\widetilde{\mathbf{U}}_{\widehat{\mathbf{H}}}^{(m)})^T$, $m \in [M]$

**4:** Find $\widehat{\mathbf{V}}^{(m)} = \widetilde{\mathbf{U}}_{\widehat{\mathbf{H}}}^{(m)}(:, 1:K_m) = \Pi_{K_m}(\widetilde{\mathbf{U}}_{\widehat{\mathbf{H}}}^{(m)})$, $m \in [M]$

**5:** For the DIMPLE model, cluster rows of $\widehat{\mathbf{V}}^{(m)}$ into $K_m$ clusters using $(1+\epsilon)$-approximate $K$-means clustering. Obtain clustering matrices $\widehat{\mathbf{Z}}^{(m)}$, $m \in [M]$

clustering matrix of the layer $m$. Hence, there are only $K_m$ distinct rows in the matrix $\mathbf{V}^{(m)}$, and clustering assignment can be obtain using step 5 of Algorithm 2.

## III. Theoretical Analysis

In this section, we study the between-layer clustering error rates of the Algorithm 1, the error of estimation of invariant subspaces for the DIMPLE-GRDPG model and and the within-layer clustering error rates of Algorithm 2. Since the clustering is unique only up to a permutation of cluster labels, denote the set of $K$-dimensional permutation functions of $[K]$ by $\aleph(K)$ and the set of $K \times K$ permutation matrices by $\mathfrak{F}(K)$. The misclassification error rate of the between-layer clustering is then given by

$$R_{BL} = (2\,L)^{-1} \min_{\mathscr{P}\in\mathfrak{F}(M)} \|\widehat{\mathbf{C}} - \mathbf{C}\,\mathscr{P}\|_F^2. \tag{18}$$

Similarly, the local community detection error in the layer of type $m \in [M]$ is

$$R_{WL}(m) = (2n)^{-1} \min_{\mathscr{P}_m\in\mathfrak{F}(K_m)} \|\widehat{\mathbf{Z}}^{(m)} - \mathbf{Z}^{(m)}\,\mathscr{P}_m\|_F^2. \tag{19}$$

Note that, since the numbering of layers is defined also up to a permutation, the errors $R_{WL}(1)$,..., $R_{WL}(M)$ should be minimized over the set of permutations $\aleph(M)$. The average error rate of the within-layer clustering is then given by

$$R_{WL} = M^{-1} \min_{\aleph(M)} \sum_{m=1}^{M} R_{WL}(m). \tag{20}$$

We shall measure the differences between the true and the estimated subspace bases matrices $\mathbf{V}^{(m)}$ and $\widehat{\mathbf{V}}^{(m)}$ using the average $\sin\Theta$ distances defined in (4). Here, again we need to seek the minimum over permutations of labels. We measure the error as $R_{S,ave}$

$$R_{S,ave} = \frac{1}{M} \min_{\aleph(M)} \sum_{m=1}^{M} \left\|\sin\Theta\left(\mathbf{V}^{(m)}, \widehat{\mathbf{V}}^{(\aleph(m))}\right)\right\|_F^2. \tag{21}$$

### A. Assumptions

In order the layers are identifiable, we assume that matrices $\mathbf{V}^{(m)}$ in (1) or $\mathbf{Z}^{(m)}$ in (2) correspond to different linear subspaces for different values of $m$. Furthermore, the performance of Algorithm 2 depends on the success of the between-layer clustering in Algorithm 1, which, in turn, relies on the fact that matrices $\mathbf{V}^{(m)}(\mathbf{V}^{(m)})^T$ in (1) or $\mathbf{Z}^{(m)}(\mathbf{Z}^{(m)})^T$ in (2), $m \in [M]$, are not too similar to each other for different values of $m$.

For the between layer clustering errors and the accuracy of the subspaces recovery, we develop our theory for the general case of the DIMPLE-GRDPG model (1). Subsequently, we derive the within-layer clustering errors for the DIMPLE model (2). Denote

$$\overline{K} = M^{-1} \sum_{m=1}^{M} K_m, \quad K = \max_{m\in[M]} K_m. \tag{22}$$

Consider matrix $\overline{\mathbf{Z}} \in \mathbb{R}^{n\times M\overline{K}}$, which is obtained as horizontal concatenation of matrices $\mathbf{V}^{(m)} \in \mathbb{R}^{n\times K_m}$, $m \in [M]$. Let the SVD of $\overline{\mathbf{Z}}$ be

$$\overline{\mathbf{Z}} = [\mathbf{V}^{(1)}|\ldots|\mathbf{V}^{(M)}] = \overline{\mathbf{U}}\,\overline{\mathbf{D}}\,\overline{\mathbf{V}}^T,$$
$$\overline{\mathbf{U}} \in \mathcal{O}_{n,r}, \overline{\mathbf{V}} \in \mathcal{O}_{M\overline{K},r}, \quad r \geq M+1. \tag{23}$$

Here, $r$ is the rank of $\overline{\mathbf{Z}}$, and $\overline{\mathbf{D}}$ is an $r$-dimensional diagonal matrix. In the case of the DIMPLE model (2), one has $\overline{\mathbf{Z}} = [\mathbf{U}_z^{(1)}|\ldots|\mathbf{U}_z^{(M)}]$. Since matrices $\mathbf{V}^{(m)}$ represent different subspaces, one has $M+1 \leq r < n$. We impose the following assumptions.

**A1.** Clusters of layers are balanced, so that there exist absolute positive constants $C_K$, $\underline{c}$ and $\bar{c}$ such that

$$C_K K \leq K_m \leq K,\ \underline{c}L/M \leq L_m \leq \bar{c}L/M,\ m \in [M], \tag{24}$$

where $L_m$ is the number of networks in the layer of type $m$. In the case of the DIMPLE model (2), local communities are balanced, so that

$$\underline{c}n/K \leq n_{k,m} \leq \bar{c}n/K, \quad k \in [K_m], m \in [M],$$

where $n_{k,m}$ is the number of nodes in the $k$-th community in the layer of type $m$.

**A2.** For some absolute constant $\kappa_0$, one has $\sigma_1(\overline{\mathbf{D}}) \leq \kappa_0\sigma_r(\overline{\mathbf{D}})$ in (23).

**A3.** The layers $\mathbf{P}^{(l)}$ of the probability tensor $\mathcal{P}$ in (1) are such that, for some absolute constant $C_\rho$ and any $l \in [L]$

$$\mathbf{P}^{(l)} = \rho_{n,l}\mathbf{P}_0^{(l)},\ \|\mathbf{P}_0^{(l)}\|_\infty = 1,\ \min_{l\in[L]} \rho_{n,l} \geq C_\rho \frac{\log n}{n}. \tag{25}$$

In the case of the DIMPLE model (2), (25) reduces to $\mathbf{B}^{(l)} = \rho_{n,l}\,\mathbf{B}_0^{(l)}$, $\|\mathbf{B}_0^{(l)}\|_\infty = 1$.

**A4.** Matrices $\mathbf{Q}^{(l)}$ in (1) are such that, for some absolute constant $C_\lambda \in (0,1)$, one has

$$\min_{l=1,\ldots.L} \left[\sigma_{K_m}\left(\mathbf{Q}^{(l)}\right)/\sigma_1\left(\mathbf{Q}^{(l)}\right)\right] \geq C_\lambda, \quad m = c(l). \tag{26}$$

In the case of the DIMPLE model, (26) appears as $\min_{l\in[L]}[\sigma_{K_m}(\mathbf{B}_0^{(l)})/\sigma_1(\mathbf{B}_0^{(l)})] \geq C_\lambda$ for $m = c(l)$.

**A5.** There exist absolute constants $\underline{c}_\rho$ and $\bar{c}_\rho$ such that

$$\underline{c}_\rho\,\rho_n \leq \rho_{n,l} \leq \bar{c}_\rho\,\rho_n \ \text{ with } \ \rho_n = (\rho_{n,1} + \cdots + \rho_{n,L})/L. \tag{27}$$

**A6.** For some absolute constant $C_{0,P}$ one has

$$\left\|\mathbf{P}_0^{(l)}\right\|_F^2 \geq C_{0,P}^2\,K^{-1}\,n^2. \tag{28}$$

Assumptions above are very common and are present in many other network papers. Specifically, Assumption **A1** is identical to Assumptions **A3** and **A4** in [25], or Assumption **A3** in [26]. Assumption **A2** is identical to Assumption **A2** in [25]. Assumption **A3** is present in majority of papers that study community detection in individual networks (see, e.g. [38]). It is required here since we rely on similarity of the sets of eigenvectors in the groups of similar layers, and, hence, need the sample eigenvectors to converge to the true ones. We believe that this assumption is **necessary** since it is also present in a majority of the multilayer network papers such as [14] and [15]. Assumption **A4** is equivalent to Assumption **A1** in [25], Assumption **A4** in [26] and an equivalent assumption in [15]. Finally, Assumption **A5**

requires that the sparsity factors are of approximately the same order of magnitude. The latter guarantees that the discrepancies between the true and the sample-based eigenvectors are similar across all layers of the network. Hypothetically, Assumption **A5** can be removed, and one can trace the impact of different scales $\rho_{n,l}$ on the clustering errors. This, however, will make clustering error bounds very complicated, so we leave this case for future investigation. Assumption **A6** postulates that matrices $\mathbf{P}_0^{(l)}$ have enough non-negligible entries. Assumption **A6** naturally holds in the case of the balanced DIMPLE model (2). Indeed, in this case, $\|\mathbf{P}_0^{(l)}\|_F^2 \geq \underline{c}^2 n^2 K^{-2} \|\mathbf{B}_0^{(l)}\|_F^2$. Due to Assumption **A3**, one has $1 = \|\mathbf{B}_0^{(l)}\|_\infty \leq \|\mathbf{B}_0^{(l)}\|$ and, therefore, by Assumptions **A1** and **A4**

$$\left\|\mathbf{B}_0^{(l)}\right\|_F^2 \geq K_m \, \sigma_{K_m}^2\left(\mathbf{B}_0^{(l)}\right) \geq C_\lambda^2 \, K_m \left\|\mathbf{B}_0^{(l)}\right\|^2 \geq C_\lambda^2 \, C_K \, K,$$

which implies $\|\mathbf{P}_0^{(l)}\|_F^2 \geq Cn^2/K$.

Note that Assumption **A3** implies that $n \to \infty$. In what follows, we assume that $L$ can grow at most polynomially with respect to $n$, specifically, that for some constant $\tau_0$

$$L \leq n^{\tau_0}, \quad 0 < \tau_0 < \infty. \tag{29}$$

Condition (29) is hardly restrictive. Indeed, [25] assume that $L \leq n$, so, in their paper, (29) holds with $\tau_0 = 1$. We allow any polynomial growth of $L$ with respect to $n$.

### B. The Between-Layer Clustering Error

Evaluation of the between-layer clustering error relies on the accuracy, measured in $\sin\Theta$ distance (4), of estimation of matrix $\mathbf{W}$ by $\widehat{\mathcal{W}}$ in Algorithm 1. The latter, due to the Davis-Kahan Theorem [39], depends on the lowest nonzero singular value of matrix $\Theta$. The structure of matrix $\Theta$ is given by the following Lemma.

*Lemma 1:* Under Assumptions **A1**–**A6**, the SVD of $\Theta$ in (12) can be written as

$$\Theta = \mathcal{V}\Lambda\mathcal{W}^T, \quad \mathcal{V} \in \mathcal{O}_{n^2,M}, \mathcal{W} = \mathbf{W}\mathbf{O}_W \in \mathcal{O}_{L,M}, \tag{30}$$

where matrix $\mathbf{W}$ is defined in (11), $\mathbf{O}_W \in \mathcal{O}_M$ and

$$\sigma_M^2(\Theta) = \sigma_{\min}^2(\Lambda) \geq (\bar{c}\,\kappa_0^4\, M)^{-1} \, \underline{c}\, C_K \, \overline{K}\, L. \tag{31}$$

Representation (30) allows one to bound above the between-layer clustering error.

*Theorem 1:* Let Assumptions **A1**–**A6** and (29) hold. Then, for any $\tau > \tau_0$, there exists a constant $C$ that depends only on $\tau$, $C_K$, $\kappa_0$, $\bar{c}$, $\underline{c}$, $\bar{c}_\rho$ and $\underline{c}_\rho$ in Assumptions **A1**–**A6**, such that the between-layer clustering error, defined in (18), satisfies

$$\mathbb{P}\left\{R_{BL} \leq C\,(n\rho_n)^{-1} K^2\right\} \geq 1 - L\,n^{-\tau} \geq 1 - n^{-(\tau-\tau_0)}. \tag{32}$$

### C. The Subspace Fitting Errors in Groups of Layers in the DIMPLE-GRDPG Model

In this section, we provide upper bounds for the divergence between matrices $\mathbf{V}^{(m)}$ and their estimators $\widehat{\mathbf{V}}^{(m)}$, $m \in [M]$. We measure their discrepancies by $R_{S,ave}$ defined in (21).

*Theorem 2:* Let Assumptions **A1**–**A6** and (29) hold, $M > 1$ and matrices $\widehat{\mathbf{V}}^{(m)}, m \in [M]$, be obtained by Algorithm 2. Let

$$\lim_{n\to\infty} (n\rho_n)^{-1} \, MK^2 = 0. \tag{33}$$

Then, for any $\tau > 0$, there exist constants $C$ that depends only on constants in Assumptions **A1**–**A6**, and a set $\Omega_{\tau,\epsilon}$ and constants $C_{\tau,\epsilon}$ that depend only on $\tau$ and $\epsilon$, such that

$$\mathbb{P}(\Omega_{\tau,\epsilon}) \geq 1 - C_{\tau,\epsilon}\, L\, n^{1-\tau}, \tag{34}$$

and, for $\omega \in \Omega_{\tau,\epsilon}$, the subspace estimation error $R_{S,ave}$ defined in (21), satisfies

$$R_{S,ave} \leq C\left\{K^5\, M \log n (nL\rho_n)^{-1} + K^5 n^{-2} + I(M>1)\, K^5 M\, \left[(n\rho_n)^{-1} + K^2\, \log^2 n (n\rho_n)^{-2}\right]\right\} \tag{35}$$

Note that, due to condition (29), if $\tau > \tau_0 + 1$, then the upper bound in (35) holds with probability at least $1 - C_{\tau,\epsilon}\, n^{-(\tau-\tau_0-1)}$.

*Remark 3:* Subspace estimation error for a homogeneous multilayer GRDPG. Consider the case when $M = 1$, so that all layers of the network can be embedded into the same invariant subspace. It follows from Theorem 2 that, for $\omega \in \Omega_{\tau,\epsilon}$, where $\Omega_{\tau,\epsilon}$ is defined in (34), one has much smaller subspace estimation error

$$R_{S,ave} \leq C\, K^5\, \left[(n\,\rho_n\, L)^{-1}\, \log n + n^{-2}\right]. \tag{36}$$

### D. The Within-Layer Clustering Error

Since the within-layer clustering for each group of layers is carried out by clustering rows of the matrices $\widehat{\mathbf{V}}^{(m)}$, the upper bound for $R_{WL}$ defined in (20) can be easily obtained as a by-product of Theorem 2. Specifically, the following statement holds.

*Corollary 1:* Let assumptions of Theorem 2 hold. Then, for any $\tau > 0$, there exists a constant $C$ that depends only on constants in Assumptions **A1**–**A6**, and $C_{\tau,\epsilon}$ which depends only on $\tau$ and $\epsilon$, such that for $\omega \in \Omega_{\tau,\epsilon}$, where $\Omega_{\tau,\epsilon}$ is defined in (34),

$$R_{WL} \leq C\left\{K^4\, M \log n (nL\rho_n)^{-1} + K^4 n^{-2} + I(M>1)\, K^4 M\, \left[(n\rho_n)^{-1} + K^2\, \log^2 n (n\rho_n)^{-2}\right]\right\}. \tag{37}$$

Note that in the case of $M = 1$, Corollary 1 yields, with high probability, that

$$R_{WL} \leq C\, K^4\, \left[(n\,\rho_n\, L)^{-1} \log n + n^{-2}\right]. \tag{38}$$

## IV. Simulation Study

In order to study performances of our methodology for various combinations of parameters, we carry out a limited simulation study with models generated from the DIMPLE and the DIMPLE-GRDPG models. We use Algorithm 1 for finding the groups of layers and Algorithms 2 and 3, respectively, for recovering the ambient subspaces in the DIMPLE-GRDPG setting, and for finding communities in groups of layers for the DIMPLE model. In addition, we carry out the simulation comparison between the DIMPLE model and the MMLSBM.

*Simulations settings:* To obtain a multilayer network that complies with our assumptions in Section III-A, we fix $n$, $L$, $M$, $K$, the sparsity parameters $c$ and $d$, the assortativity parameter $w$, and the Dirichlet parameter $\alpha$ used for generating a DIMPLE-GRDPG network. We use the multinomial distribution with equal probabilities $1/M$ to assign group memberships to individual networks.

In the case of the DIMPLE model, we generate $K$ communities in each of the groups of layers using the multinomial distribution with equal probabilities $1/K$. In this manner, we obtain community assignment matrices $\mathbf{Z}^{(m)}$, $m \in [M]$, in each layer $l$ with $c(l) = m$, where $c : [L] \to [M]$ is the layer assignment function. Next, we generate the entries of $\boldsymbol{B}^{(l)}$, $l \in [L]$, as uniform random numbers between $c$ and $d$, and then multiply all the non-diagonal entries of those matrices by $w$. In this manner, if $w < 1$ is small, then the network is strongly assortative, i.e., there is a higher probability for nodes in the same community to connect. If $w > 1$ is large, then the network is disassortative, i.e., the probability of connection for nodes in different communities is higher than for nodes in the same community. Finally, since entries of matrices $\boldsymbol{B}^{(l)}$ are generated at random, when $w$ is close to one, the networks in all layers are neither assortative or disassortative. After the community assignment matrices $\mathbf{Z}^{(m)}$ and the block probability matrices $\boldsymbol{B}^{(l)}$ have been obtained, we construct the probability tensor $\mathcal{P}$ with layers $\mathcal{P}(:,:,l) = \mathbf{Z}^{(m)}\boldsymbol{B}^{(l)}(\mathbf{Z}^{(m)})^T$, where $m = c(l)$, $l \in [L]$.

In the case of the DIMPLE-GRDPG setting, we obtain matrices $\mathbf{X}^{(m)} \in [0,1]^{n\times K}$, $m \in [M]$, with independent rows, generated using the Dirichlet distribution with parameter $\alpha$. We obtain matrices $\boldsymbol{B}^{(l)}$, in exactly the same manner as in the case of the DIMPLE model and construct $\mathcal{P}$ with layers $\mathcal{P}(:,:,l) = \mathbf{X}^{(m)}\boldsymbol{B}^{(l)}(\mathbf{X}^{(m)})^T$, where $m = c(l)$, $l \in [L]$. In this case, the matrices $\mathbf{V}^{(m)}$ are obtained from the SVD $\mathbf{X}^{(m)} = \mathbf{V}^{(m)}\Lambda_X^{(m)}\mathbf{W}_X^{(m)}$ of $\mathbf{X}^{(m)}$. Matrices $\mathbf{Q}^{(l)}$ are defined as $\mathbf{Q}^{(l)} = \Lambda_X^{(m)}\mathbf{W}_X^{(m)}\boldsymbol{B}^{(l)}(\mathbf{W}_X^{(m)})^T\Lambda_X^{(m)}$ in (1), $l \in [L]$.

After the probability tensor $\mathcal{P}$ has been generated, the layers $\mathbf{A}^{(l)}$ of the adjacency tensor $\mathcal{A}$ are obtained as symmetric matrices with zero diagonals and independent Bernoulli entries $\mathbf{A}^{(l)}(i,j)$ for $1 \le i < j \le n$. Subsequently, we use Algorithm 1 for finding the groups of layers for both models, followed by Algorithm 2 for estimating matrices $\mathbf{V}^{(m)}$ in the case of the DIMPLE-GRDPG network, or clustering nodes in each group of layers of the network into communities for the DIMPLE model. In both cases, we have two sets of simulations, one with fixed $L$ and varying $n$, another with the fixed $n$ and varying $L$. In all simulations, we set $M = 3$ and $K_m = 3$ for $m = 1, 2, 3$, and study two sparsity scenarios, $c = 0$, $d = 0.8$ or $c = 0$, $d = 0.5$, with four values of assortativity parameter $w = 0.6, 0.8, 1.0$ and $1.2$. In all simulations, we set $\alpha = 0.1$. We report the average between-layer clustering errors $R_{BL}$ defined in (18), and also the average within-layer clustering error $R_{WL}$ defined in (20) in the case of the DIMPLE setting and the average $\sin\Theta$ distance $R_{S,ave}$ defined in (21) between the true and the estimated subspaces in the case of the DIMPLE-GRDPG network. We first present simulations results for the DIMPLE model followed by the study of the DIMPLE-GRDPG model.

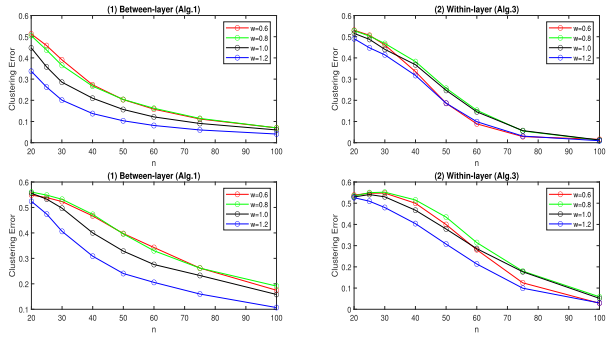


Fig. 1. Between-layer clustering error rates of Algorithm 1 (left) and the within-layer error rates of Algorithms 2 (right), averaged over 500 simulation runs, for the **DIMPLE** model with $c = 0, d = 0.8$ (top) and $c = 0, d = 0.5$ (bottom), $L = 50$ and $n = 20, 25, 30, 40, 50, 60, 75, 100$.

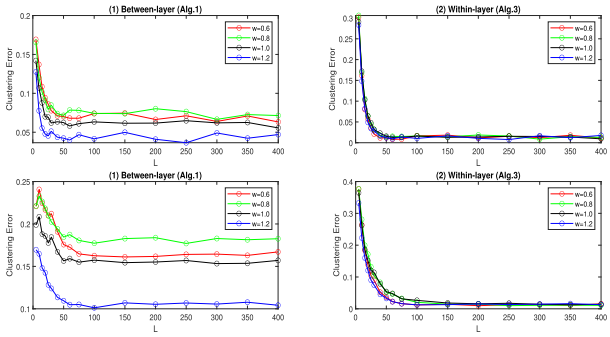


Fig. 2. Between-layer clustering error rates of Algorithm 1 (left) and the within-layer error rates of Algorithms 2 (right), averaged over 500 simulation runs, for the **DIMPLE** model with $c = 0, d = 0.8$ (top) and $c = 0, d = 0.5$ (bottom), $n = 100$ and $L = 5, 10, 15, 20, 25, 30, 40, 50, 60, 75, 100, 150, 200, 250, 300, 350, 400$.

*Simulations results:* Both estimation and clustering are harder when a network is more sparse, therefore, all errors are smaller when $d = 0.8$ (top panels) than when $d = 0.5$ (bottom). Figs. 1–4 show that the value of the assortativity parameter does not play a significant role in the between-layer clustering. Indeed, as the left panels in all figures show, the smallest between-layer clustering errors occur for $w = 1.2$ followed by $w = 1.0$. The latter confirms that the difficulty of the between-layer clustering is predominantly controlled by the sparsity of the network. The results are somewhat different for the community detection errors and the subspace estimation errors in, respectively, the DIMPLE and the DIMPLE-GRDPG models. Indeed, as the right panels in Figs. 1–4 show, the smallest errors occur

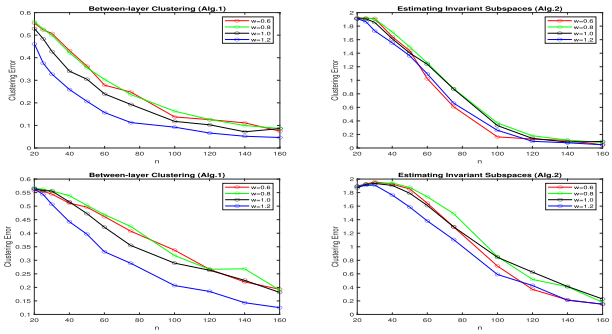


Fig. 3. Between-layer clustering error rates of Algorithm 1 (left) and the $\sin\Theta$ distances $R_{S,ave}$ of Algorithms 2 (right), averaged over 100 simulation runs, for the **DIMPLE-GRDPG** model with $\alpha = 0.1$, $c = 0$, $d = 0.8$ (top) and $c = 0, d = 0.5$ (bottom), $L = 50$ and $n = 20, 25, 30, 40, 50, 60, 75, 100, 120, 140, 160$.

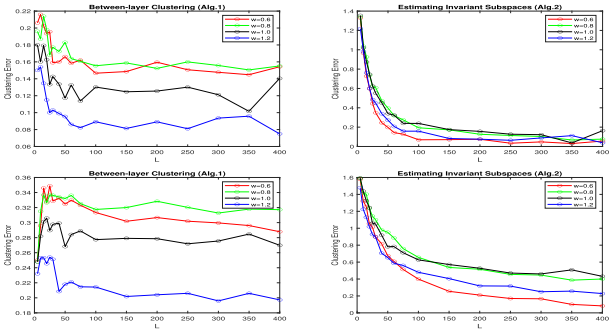


Fig. 4. Between-layer clustering error rates of Algorithm 1 (left) and the $\sin\Theta$ distances $R_{S,ave}$ of Algorithms 2 (right), averaged over 100 simulation runs, for the **DIMPLE-GRDPG** model with $\alpha = 0.1$, $c = 0, d = 0.8$ (top) and $c = 0, d = 0.5$ (bottom), $n = 100$ and $L = 5, 10, 15, 20, 25, 30, 40, 50, 60, 75, 100, 150, 200, 250, 300, 350, 400$.

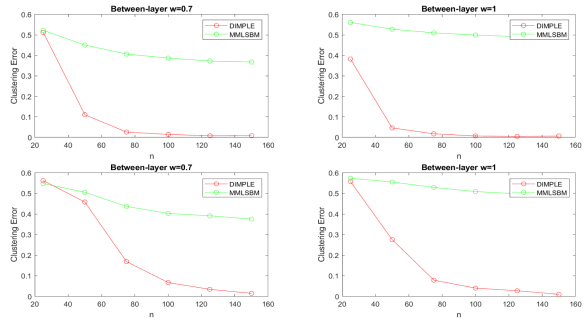


Fig. 5. Between-layer clustering error rates of Algorithm 1 and Alternative Minimization Algorithm of [26]. Data are generated using DIMPLE model with $L = 50$, $c = 0$, $d = 0.8$ (top) and $c = 0$, $d = 0.5$ (bottom), and $w = 0.7$ (left panel) or $w = 1$ (right panel).

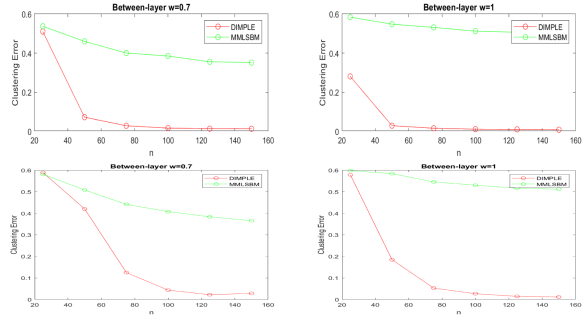


Fig. 6. Between-layer clustering error rates of Algorithm 1 and Alternative Minimization Algorithm of [26]. Data are generated using DIMPLE model with $L = 100$, $c = 0$, $d = 0.8$ (top) and $c = 0$, $d = 0.5$ (bottom), $n = 20, 40, 60, 80, 100, 120, 140, 160$ and $w = 0.7$ (left panel) or $w = 1$ (right panel).

in the more assortative/disassortative models with $w = 0.6$ and $w = 1.2$.

One can see from Figs. 1 and 3 that, when $n$ grows, all errors decrease. The influence of $L$ on the error rates is more complex. As Theorem 1 implies, the between-layer clustering errors are of the order $(n\rho_n)^{-1}$ for fixed values of $M$ and $K$. This agrees with the left panels in Figs. 2 and 4 where curves exhibit constant behavior for when $L$ grows (small fluctuations are just due to random errors). For the right panels in Figs. 2 and 4 this, however, happens only when $L$ is relatively large.

The explanation for such behavior lies in the fact that the between-layer clustering error (corresponding to the left panels in Figs. 2 and 4) is of the order $K^2\,(n\rho_n)^{-1}$ and is independent of $L$. On the other hand, for fixed $K$ and $M$, the errors $R_{WL}$ and $R_{S,ave}$ (corresponding to the right panels in, respectively, Figs. 2 and 4) are of the order $(n\,\rho_n)^{-1} + \log n\,(n\,\rho_n\,L)^{-1}$. While $L$ is small the second term is dominant but, as $L$ grows. the first term becomes dominant and the errors stop declining as $L$ grows.

*The DIMPLE model versus the MMLSBM:* In this paper, we consider the DIMPLE model, which is a more general model than the MMLSBM. Specifically, the MMLSBM has only $M$ types of layers in the tensor and, therefore, results in a low rank tensor. On the other hand, all tensor layers in the DIMPLE model can be different and, therefore, the tensor is not of low rank. In this section, we carry out a limited simulation study, the purpose of which is to convince a reader that, while our algorithms work in the case of the MMLSBM, the algorithms designed for the MMLSBM produce poor results when data are generated according to the DIMPLE models.

For facilitation of such comparisons, we generate data for the DIMPLE model in exactly the same manner as it is done above. In order to generate data according to the MMLSBM, we note that the main difference between the MMLSBM and the DIMPLE model is that, in MMLSBM, one has only $M$ distinct matrices $\boldsymbol{B}^{(l)}$, since $\boldsymbol{B}^{(l)} = \boldsymbol{B}^{(c(l))}$, $l \in [L]$. Hence, we generate $M$ matrices $\boldsymbol{B}^{(m)}$, $m \in [M]$, and then set $\boldsymbol{B}^{(l)} = \boldsymbol{B}^{(c(l))}$, $l \in [L]$. We present only the between layer clustering errors since the within-layer clustering in the MMLSBM and the DIMPLE model can be carried out in a similar way. We compare the performances of Algorithm 1 in this paper with the Alternative Minimization Algorithm (ALMA) of [26]. This choice, rather than the TWIST algorithm of [25], is motivated by the fact that in [26] AMLA was shown to be more precise than TWIST in majority of situations. Figs. 5–8 exhibit results of application of Algorithm 1 and ALMA of [26] for $K = 5$, $M = 3$ and various values of $L$, $n$, $c$, $d$ and $w$.

Specifically, Figs. 5 and 6 show that, when data are generated according to the DIMPLE model, Algorithm 1 in our paper allows to reliably separate layers of the network into $M$ types,

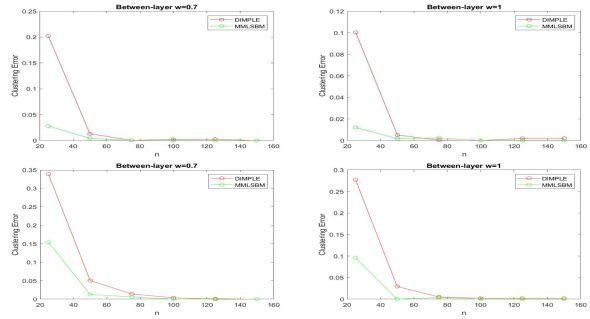


Fig. 7. Between-layer clustering error rates of Algorithm 1 and Alternative Minimization Algorithm of [26]. Data are generated using MMLSBM with $L = 50$, $c = 0$, $d = 0.8$ (top) and $c = 0$, $d = 0.5$ (bottom), $n = 20, 40, 60, 80, 100, 120, 140, 160$ and $w = 0.7$ (left panel) or $w = 1$ (right panel).

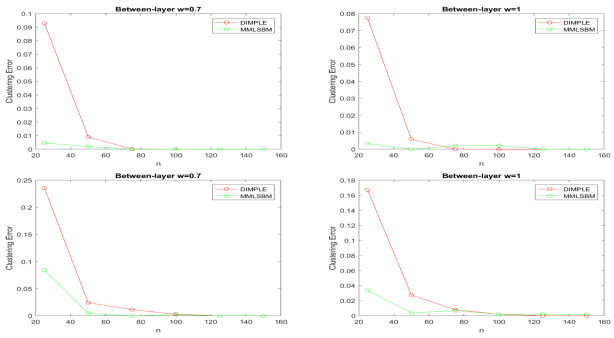


Fig. 8. Between-layer clustering error rates of Algorithm 1 and Alternative Minimization Algorithm of [26]. Data are generated using MMLSBM with $L = 100, c = 0, d = 0.8$ (top) and $c = 0, d = 0.5$ (bottom), and $w = 0.7$ (left panel) or $w = 1$ (right panel).

while ALMA fails to do so. The reason for this is that ALMA expects the matrices of probabilities to be identical in those layers, although, in reality, they are not. As a result, when $n$ grows, the clustering errors do not tend to zero but just flatten.

Figs. 7 and 8 exhibit results of application of Algorithm 1 and ALMA when data are generated according to the MMLSBM. As it is expected, for small values of $n$, ALMA leads to a better clustering precision. The latter is due to the fact that Algorithm 1 relies on the SVDs of the layers of the adjacency tensor $\mathcal{A}$, that are not reliable for small values of $n$. In addition, Algorithm 1 cannot take into account that the probability tensor is of a low rank since this is not true for the DIMPLE model. However, these advantages become less and less significant as $n$ grows. As Figs. 7 and 8 show, both algorithms have similar clustering precision for larger values of $n$, specifically, for $n \geq n_0$, where $n_0$ is between 60 and 100, depending on a particular simulations setting.

## V. Application to the Real World Data

In this section, we consider applications of the DIMPLE and the DIMPLE-GRDPG models to real-life data, and its comparison with the MMLSBM. Note that since the GRDPG includes all other block network models as its particular cases, the latter obviates making those choices. Algorithm 1 works equally well for a multilplex network equipped with any type of such model.

In our examples, however, the DIMPLE model with its SBM-imposed structures provided better descriptions of the organization of layers in each group than its GRDPG-based DIMPLE-GRDPG counterpart. This allowed us to compare data analyses under the DIMPLE model with the similar analyses under the simpler MMLSBM. MMLSBM is the only model which we can use for comparison with our model since all other multiplex network models assume that all layers follow the same community structure.

In what follows, we examine two real data sets: the Global Flights Network Data and the Worldwide Food Trading Networks data. Both examples demonstrate that application of our much more flexible DIMPLE model leads to much more intuitive description of the data which, in our view, validates the introduction of the model and its analysis in this paper.

### A. Global Flights Network Data

In this subsection, we applied our clustering algorithms to the Global Flights Network data collected by the OpenFlights. As of June 2014, the OpenFlights Database contains 67663 routes between 3321 airports on 548 airlines spanning the globe. It is available at https://openflights.org/data.html#airport.

These data can be modeled as a multiplex network, in which layers represent different airlines, nodes are airports where airlines depart and land, and edges at each layer represent existing routes of a specific airline company between two airports. To avoid sparsity, we selected 224 airports, where over 150 airline companies have rights to depart and land in. Furthermore, we chose 81 airlines that have at least 240 routes between those airports, constructing a network with 224 nodes and 81 layers. We scrambled the 81 layers and applied Algorithm 1 for the between-layer clustering. As it is described in Section II, we found the number of communities in each of the layers by using the ScreeNOT technique of [37].

Choosing the number of groups of layers for this network is, however, a very different story. In our experiments, we tried various values of $M$ ranging from $M = 2$ to $M = 7$. Each of this arrangements makes perfect sense, and, as $M$ grows, the partitions of the airlines into groups exhibit higher and higher level of differentiation. For example, if $M = 2$, one group consists of the airlines based in China, South Korea and Japan while another group consists of all other airlines. When $M = 3$, the latter group splits into the airlines based in Europe/Middle East and Americas. When $M = 4$, the airlines are further partitioned into the ones based in China (group 1), in Asia and Australia (group 2), in Europe and middle East (group 3) and in US, Canada, Mexico and South America (group 4). At $M = 5$, the airlines based in Europe and Middle East separate into two clusters, one primarily based in Northern Europe and another containing the rest. At $M = 6$, the airlines based in South America form a new group. At $M = 7$, the three airlines from India separate into a distinct cluster.

In this paper, we display results of the between-layer clustering for $M = 4$ in Table I. We use the consensus ambient dimension $K = 3$ for each of the groups of layers. It is easy to see that the airlines are naturally grouped by geographical areas from where the flights are originated. Group 1 is constituted

TABLE I
AIRLINES GROUPS, FOR THE DIMPLE MODEL, OBTAINED USING ALGORITHM 1 WITH $K = 3$ AND $M = 4$

| Airlines Groups under the DIMPLE-GDPG Model | | | |
|---|---|---|---|
| **Group 1** | | **Group 2** | |
| China | Hainan Airlines | New Zealand | Air New Zealand |
| China | Air China | Republic of Korea | Korean Air |
| China | Sichuan Airlines | Singapore | Singapore Airlines |
| China | Shenzhen Airlines | Australia | Qantas |
| China | China Southern Airlines | Vietnam | Vietnam Airlines |
| China | Shandong Airlines | India | Air India Limited |
| China | China Eastern Airlines | India | IndiGo Airlines |
| China | Xiamen Airlines | Australia | Virgin Australia |
| Japan | Japan Air System | South Africa | South African Airways |
| **Group 3** | | Indonesia | Garuda Indonesia |
| Germany | Lufthansa | Republic of Korea | Asiana Airlines |
| Russia | Ural Airlines | Malaysia | Malaysia Airlines |
| Switzerland | Swiss International Air Lines | India | Jet Airways |
| Morocco | Royal Air Maroc | Japan | Japan Airlines |
| Norway | Norwegian Air Shuttle | Japan | All Nippon Airways |
| Ireland | Ryanair | Qatar | Qatar Airways |
| Turkey | Turkish Airlines | Saudi Arabia | Saudi Arabian Airlines |
| Greece | Aegean Airlines | United Arab Emirates | Emirates |
| Algeria | Air Algerie | United Arab Emirates | Etihad Airways |
| Ethiopia | Ethiopian Airlines | **Group 4** | |
| United Kingdom | Jet2.com | United States | JetBlue Airways |
| United Kingdom | Flybe | United States | US Airways |
| Russia | Transaero Airlines | United States | Alaska Airlines |
| Germany | Condor Flugdienst | United States | Southwest Airlines |
| Germany | TUIfly | United States | Delta Air Lines |
| Sweden | Scandinavian Airlines | United States | AirTran Airways |
| Portugal | TAP Portugal | United States | Spirit Airlines |
| France | Transavia France | United States | United Airlines |
| United Kingdom | British Airways | United States | American Airlines |
| Russia | S7 Airlines | United States | Frontier Airlines |
| Ireland | Aer Lingus | Canada | Air Canada |
| Germany | Germanwings | Canada | WestJet |
| Egypt | Egyptair | Mexico | AeroMexico |
| Austria | Austrian Airlines | Chile | LAN Airlines |
| Spain | Iberia Airlines | Brazil | TAM Brazilian Airlines |
| Germany | Air Berlin | South America | Avianca |
| Italy | Alitalia | Netherlands | KLM Royal Dutch Airlines |
| Hungary | Wizz Air | France | Air France |
| Finland | Finnair | | |
| Russia | Aeroflot | | |
| France | Air Bourbon | | |
| Netherlands | Transavia Holland | | |
| United Kingdom | easyJet | | |

by Chinese airline and one Japanese airline which has flights predominantly in Far East. Group 2 consists of airlines that belong to countries in Asia, such as India, Japan, South Korea and Vietnam, Australia and New Zealand, and few big airlines in Gulf States (Saudi Arabia, United Arab Emirates, Qatar) that have a large number of flights to both Asia and Australia. Group 3 is formed by airlines originated from Europe and North Africa. Group 4 is comprised of airlines that fly in or from North or South America. Not surprisingly, this group includes two big European airlines, KLM and Air France, since those airlines are members of the SkyTeam alliance and share many flights originated in USA with Delta airlines.

Furthermore, we compare the clustering assignments with the ones obtained by the ALMA algorithm designed for the MMLSBM. To this end, we applied ALMA algorithm of [26] for the layer clustering, with the same parameters $M = 4$ and $K = 3$. Results are presented in Table II. It is easy to see that while the DIMPLE model ensures a logical geography-based partition of the airlines, the MMLSBM does not. Indeed, the MMLSBM lumps almost all airlines into Group 1, placing few Chinese airlines into Group 2, few United States owned airlines together with Air France, Alitalia and KLM into Group 3, and Ryanair (Ireland), Transavia and Air Bourbon (France), easyJet and Jet2.com (United Kingdom) into Group 4. On the contrary, Algorithm 1 associated with the DIMPLE model delivers four balanced (similar in size) groups. This is due to the fact that MMLSBM groups airlines by the volume of operation rather than the structure of roots.

We also partitioned airports in each of the groups of airlines into communities. Results are presented in Fig. 9.

### B. *Worldwide Food Trading Network Data*

We apply our algorithms to the Worldwide Food Trading Networks data collected by the Food and Agriculture Organization of the United Nations. The data have been described in [40], and it is available at https://www.fao.org/faostat/en/#data/TM. The data includes export/import trading volumes among 245 countries for more than 300 food items. These data can be modeled as a multiplex network, in which layers represent

TABLE II
AIRLINES GROUPS, FOR THE MMLSBM, OBTAINED USING ALMA ALGORITHM OF [26] WITH $K = 3$ AND $M = 4$

| Airlines Groups under the MMLSBM | | | |
|---|---|---|---|
| **Group 1** | | **Group 2** | |
| Japan | Japan Air System | China | Hainan Airlines |
| China | Sichuan Airlines | China | Air China |
| China | Shandong Airlines | China | Shenzhen Airlines |
| China | Xiamen Airlines | China | China Southern Airlines |
| Republic of Korea | Korean Air | China | China Eastern Airlines |
| Singapore | Singapore Airlines | | |
| Vietnam | Vietnam Airlines | **Group 3** | |
| India | Air India Limited | France | Air France |
| United States | US Airways | United States | Delta Air Lines |
| Australia | Qantas | United States | AirTran Airways |
| Mexico | AeroMexico | United States | Southwest Airlines |
| India | IndiGo Airlines | United States | American Airlines |
| South Africa | South African Airways | Netherlands | KLM Royal Dutch Airlines |
| Indonesia | Garuda Indonesia | Italy | Alitalia |
| Republic of Korea | Asiana Airlines | | |
| Saudi Arabia | Saudi Arabian Airlines | **Group 4** | |
| Hong Kong | Cathay Pacific | France | Transavia France |
| South America | Avianca | France | Air Bourbon |
| Japan | Japan Airlines | United Kingdom | Jet2.com |
| Qatar | Qatar Airways | United Kingdom | easyJet |
| Australia | Virgin Australia | Ireland | Ryanair |
| Japan | All Nippon Airways | | |
| Malaysia | Malaysia Airlines | **Group 1: Continuation** | |
| India | Jet Airways | Canada | WestJet |
| United Arab Emirates | Etihad Airways | United Arab Emirates | Emirates |
| Germany | Lufthansa | Russia | Ural Airlines |
| Turkey | Pegasus Airlines | Morocco | Royal Air Maroc |
| Switzerland | Swiss International Airlines | Turkey | Turkish Airlines |
| Norway | Norwegian Air Shuttle | Ethiopia | Ethiopian Airlines |
| Greece | Aegean Airlines | Algeria | Air Algerie |
| United Kingdom | Flybe | Germany | Condor Flugdienst |
| Germany | TUIfly | Sweden | Scandinavian Airlines |
| Portugal | TAP Portugal | United Kingdom | British Airways |
| Russia | S7 Airlines | Austria | Austrian Airlines |
| Ireland | Aer Lingus | Spain | Iberia Airlines |
| Germany | Germanwings | Russia | Aeroflot |
| Egypt | Egyptair | Germany | Air Berlin |
| Hungary | Wizz Air | Russia | Transaero Airlines |
| Finland | Finnair | United States | Alaska Airlines |
| Netherlands | Transavia Holland | Brazil | TAM Brazilian Airlines |
| United States | JetBlue Airways | United States | Spirit Airlines |
| Chile | LAN Airlines | Canada | Air Canada |
| New Zealand | Air New Zealand | United States | Frontier Airlines |
| United States | United Airlines | | |

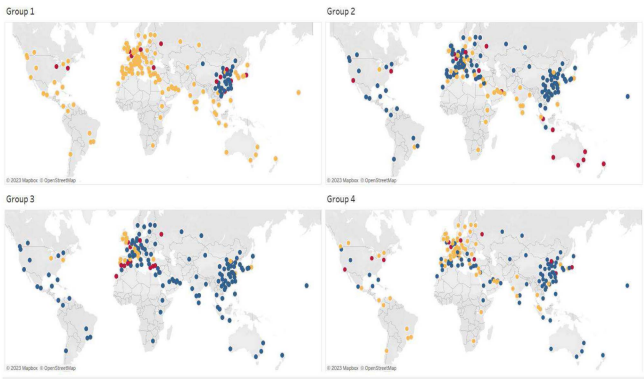


Fig. 9. Communities for the four airlines groups. Group 1: airlines originated in China. Group 2: airlines originated in Asia, Australia,New Zealand, and Gulf States. Group 3: airlines originated in Europe and North Africa. Group 4: airlines originated in North or South America.

different products, nodes are countries, and edges at each layer represent trading relationships of a specific food product among countries. A part of the data set was analyzed in [25] and [26].

Similarly to [25] and [26], we used data for the year 2010. We start with pre-processing the data by adding the export and import volumes for each pair of countries in each layer of the network, to produce undirected networks that fit in our model. To avoid sparsity, we select 104 countries, whose total trading volumes are higher than the median among all countries. We choose 58 meat/dairy and fruit/vegetable items and constructed a network with 104 nodes and 58 layers.

While pre-processing the data, we observe that global trading patterns are different for the meat/dairy and the fruit/vegetable groups. Specifically, the trading volumes in meat/dairy group are much smaller than the trading volumes in the fruit/vegetable group. For this reason, we choose the thresholds that keep similar

| | | |
|---|---|---|
| Meat Group Cluster 1 | Bacon and ham | Meat, pig |
| | Butter, cow milk | Meat, pig sausages |
| | Fat, pigs | Meat, pork |
| | Eggs, liquid | Offals, pigs, edible |
| | Meat, chicken | Meat, beef, preparations |
| | Meat, cattle | Meat, turkey |
| | Meat, cattle, boneless (beef & veal) | Milk, whole dried |
| | Meat, pig, preparations | Meat, sheep |
| | Offals, edible, cattle | Meat, nes |
| | Meat, chicken, canned | Meat, duck |
| | Tallow | Offals, sheep,edible |
| Fruit/Vegetables Group Cluster 2 | Apples | Grapefruit (inc. pomelos) |
| | Avocados | Peaches and nectarines |
| | Cabbages and other brassicas | Plums and sloes |
| | Carrots and turnips | Tangerines, mandarins, clementines |
| | Cauliflowers and broccoli | Watermelons |
| | Cherries | Strawberries |
| | Chillies and peppers, green | Tomatoes |
| | Cucumbers and gherkins | Beans, green |
| | Figs dried | Fruit, fresh nes |
| | Kiwi fruit | Fruit, tropical fresh nes |
| | Oranges | Asparagus |
| | Papayas | Cassava dried |
| | Maize, green | Pineapples |
| | Persimmons | Leeks, other alliaceous vegetables |
| | Vegetables, fresh nes | Juice, pineapple, concentrated |
| | Spinach | Peas, green |
| | Sweet potatoes | Onions, shallots, green |
| | Roots and tubers, nes | Mangoes, mangosteens, guavas |

Fig. 10. Results of clustering of food networks layers into $M = 2$ clusters by Algorithm 1 in the paper.

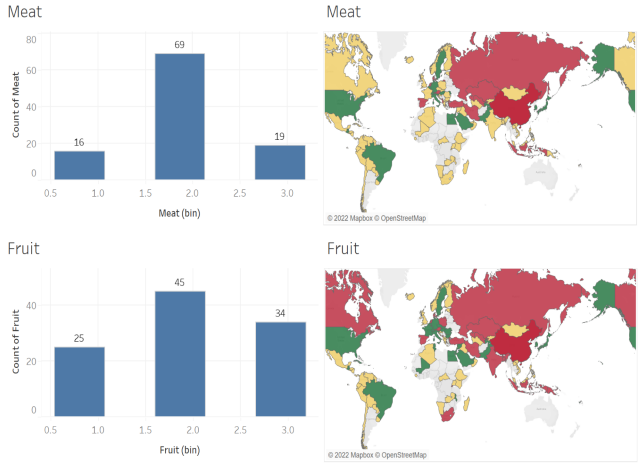


Fig. 11. Trading communities for the meat/dairy (top) and the fruit/vegetable (bottom) groups. Left panels: community sizes; right panels: community memberships.

sparsity levels for the adjacency matrices. In particular, we set threshold to be equal to 1 unit for the meat/dairy group and 300 units for the fruit/vegetable group, and draw an edge between two nodes (countries) if the total trading volume between them is at or above the threshold.

We scramble the 58 layers and apply Algorithm 1 for the between-layer clustering. Since the food items consist of a meat/dairy and a fruit/vegetable group, we set $M = 2$. We estimated the ambient dimension in each layer using the ScreeNOT procedure in [37], and obtained the between-layer clustering assignments using Algorithm 1. Results of the between-layer clustering are presented in Fig. 10. As it is evident from Fig. 10, Algorithm 1 separates the food items into the meat/dairy and the fruit/vegetable groups.

In order to describe the structures of the meat/dairy and the fruit/vegetable groups of layers, we furthermore assume that the layers of the network follow the SBMs which allows to investigate the communities of countries that form trade clusters in each of the two layers. Using the consensus $K = 3$ for the number of layers in each of the two groups, we use Algorithm 2 and exhibit results of the within-layer clustering in Fig. 11. The left panels in Fig. 11 show the number of nodes (countries) in communities 1,2 and 3 in the meat/dairy and the fruit/vegetable group, respectively. The right panels in Fig. 11 project those countries onto the world map. Here, the red color is used for community 1, the yellow color for community 2 and the green color for for community 3. Since we only select 104 countries to be a part of the network, some regions in the map are colored grey.

Additionally, in order to justify application of the DIMPLE model, we also carry out data analysis assuming that data were generated using the MMLSBM. Specifically, we applied ALMA algorithm of [26] to the layer clustering with parameters $M = 2$ and $K = 3$. Results are presented in Fig. 12. It is easy to see that ALMA algorithm places some of the meat/dairy items into the fruit/vegetable group. We believe that this is due to the fact that MMLSBM is sensitive to the probabilities of connections rather than connection patterns.

| | | |
|---|---|---|
| Cluster1 | Butter, cow milk | Meat, pig |
| | Eggs, liquid | Meat, pig sausages |
| | Meat, chicken | Meat, pork |
| | Meat, cattle | Meat, beef, preparations |
| | Meat, cattle, boneless (beef & veal) | Meat, turkey |
| | Meat, pig, preparations | Milk, whole dried |
| | Offals, edible, cattle | Meat, sheep |
| | Meat, chicken, canned | Meat, nes |
| | Tallow | |
| Cluster 2 | Bacon and ham | Tomatoes |
| | Fat, pigs | Beans, green |
| | Offals, pigs, edible | Fruit, fresh nes |
| | Meat, duck | Fruit, tropical fresh nes |
| | Offals, sheep,edible | Asparagus |
| | Apples | Cassava dried |
| | Avocados | Pineapples |
| | Cabbages and other brassicas | Leeks, other alliaceous vegetables |
| | Carrots and turnips | Juice, pineapple, concentrated |
| | Cauliflowers and broccoli | Peas, green |
| | Cherries | Onions, shallots, green |
| | Chillies and peppers, green | Mangoes, mangosteens, guavas |
| | Cucumbers and gherkins | Papayas |
| | Figs dried | Maize, green |
| | Kiwi fruit | Persimmons |
| | Oranges | Vegetables, fresh nes |
| | Grapefruit (inc. pomelos) | Spinach |
| | Peaches and nectarines | Sweet potatoes |
| | Plums and sloes | Roots and tubers, nes |
| | Tangerines, mandarins, clementines | Strawberries |
| | Watermelons | |

Fig. 12. Results of clustering of food networks layers into $M = 2$ clusters by ALMA algorithm of [26].

## VI. Discussion

In this paper, we introduce the GDPG-equipped DIMPLE-GDPG multiplex network model where layers can be partitioned into groups with similar ambient subspace structures while the matrices of connections probabilities can be all different. In the common case when each layer follows the SBM, the latter reduces to the DIMPLE model, where community affiliations are common for each group of layers while the matrices of block connection probabilities vary from one layer to another. Our real data examples in Section V show that our models deliver more logical description of data than the MMLSBM, due to the flexibility of the DIMPLE and DIMPLE-GDPG models.

If $M = 1$, the DIMPLE-GDPG reduces to COSIE model, and we believe that our paper provides some improvements due to employment of a different algorithm for the matrix $\mathbf{V}$ estimation. The detail comparison of convergence rates can be found in Section A

## REFERENCES


[1] M. Kivela, A. Arenas, M. Barthelemy, J. P. Gleeson, Y. Moreno, and M. A. Porter, "Multilayer networks," *J. Complex Netw.*, vol. 2, no. 3, pp. 203–271, Jul. 2014, doi: 10.1093/comnet/cnu016.

[2] A. Aleta and Y. Moreno, "Multilayer networks in a nutshell," *Annu. Rev. Condens. Matter Phys.*, vol. 10, no. 1, pp. 45–62, Mar. 2019, doi: 10.1146/annurev-conmatphys-031218-013259.

[3] D. Durante, N. Mukherjee, and R. C. Steorts, "Bayesian learning of dynamic multilayer networks," *J. Mach. Learn. Res.*, vol. 18, no. 43, pp. 1–29, 2017. [Online]. Available: http://jmlr.org/papers/v18/16-391.html

[4] S. Han and D. B. Dunson, "Multiresolution tensor decomposition for multiple spatial passing networks," 2018, *arXiv:1803.01203*.

[5] T.-C. Kao and M. A. Porter, "Layer communities in multiplex networks," *J. Stat. Phys.*, vol. 173, no. 3/4, pp. 1286–1302, Aug. 2017, doi: 10.1007/s10955-017-1858-z.

[6] P. W. MacDonald, E. Levina, and J. Zhu, "Latent space models for multiplex networks with shared structure," *Biometrika*, vol. 109, no. 3, Nov. 2021, doi: 10.1093/biomet/asab058.

[7] F. Lorrain and H. C. White, "Structural equivalence of individuals in social networks," *J. Math. Sociol.*, vol. 1, no. 1, pp. 49–80, 1971, doi: org/10.1080/0022250X.1971.9989790.

[8] S. C. Olhede and P. J. Wolfe, "Network histograms and universality of blockmodel approximation," *Proc. Nat. Acad. Sci. USA*, vol. 111, no. 41, pp. 14722–14727, 2014. [Online]. Available: https://www.pnas.org/content/111/41/14722

[9] O. Sporns, "Graph theory methods: Applications in brain networks," *Dialogues Clin. Neurosci.*, vol. 20, no. 2, pp. 111–121, 2018.

[10] N. A. Crossley et al., "Cognitive relevance of the community structure of the human brain functional coactivation network," *Proc. Nat. Acad. Sci. USA*, vol. 110, no. 28, pp. 11583–11588, 2013.

[11] J. Faskowitz, X. Yan, X.-N. Zuo, and O. Sporns, "Weighted stochastic block models of the human connectome across the life span," *Sci. Rep.*, vol. 8, no. 1, 2018, Art. no. 12997. [Online]. Available: https://app.dimensions.ai/details/publication/pub.1106343698 and https://www.nature.com/articles/s41598-018-31202-1.pdf

[12] C. Nicolini, C. Bordier, and A. Bifone, "Community detection in weighted brain connectivity networks beyond the resolution limit," *Neuroimage*, vol. 146, pp. 28–39, 2017.

[13] P. Rubin-Delanchy, J. Cape, M. Tang, and C. E. Priebe, "A statistical interpretation of spectral embedding: The generalised random dot product graph," *J. Roy. Statist. Soc., Ser. B*, vol. 84, pp. 1446–1473, 2022.

[14] J. Arroyo, A. Athreya, J. Cape, G. Chen, C. E. Priebe, and J. T. Vogelstein, "Inference for multiple heterogeneous networks with a common invariant subspace," *J. Mach. Learn. Res.*, vol. 22, no. 142, pp. 1–49, 2021. [Online]. Available: http://jmlr.org/papers/v22/19-558.html

[15] R. Zheng and M. Tang, "Limit results for distributed estimation of invariant subspaces in multiple networks inference and PCA," 2022, *arXiv:2206.04306*.

[16] P. Brodka, A. Chmiel, M. Magnani, and G. Ragozini, "Quantifying layer similarity in multiplex networks: A systematic study," *Roy. Soc. Open Sci.*, vol. 5, no. 8, 2018, Art. no. 171747. [Online]. Available: https://royalsocietypublishing.org/doi/abs/10.1098/rsos.171747

[17] P. Mercado, A. Gautier, F. Tudisco, and M. Hein, "The power mean laplacian for multilayer graph clustering," in *Proc. 21st Int. Conf. Artif. Intell. Statist. Mach. Learn. Res.*, 2018, vol. 84, pp. 1828–1838.

[18] S. Bhattacharyya and S. Chatterjee, "General community detection with optimal recovery conditions for multi-relational sparse networks with dependent layers," 2020, *arXiv:2004.03480*.

[19] J. Lei and K. Z. Lin, "Bias-adjusted spectral clustering in multi-layer stochastic block models," *J. Amer. Stat. Assoc.*, vol. 118, no. 544, pp. 2433–2445, 2023.

[20] S. Paul and Y. Chen, "Consistent community detection in multi-relational data through restricted multi-layer stochastic blockmodel," *Electron. J. Statist.*, vol. 10, no. 2, pp. 3807–3870, 2016, doi: 10.1214/16-EJS1211.

[21] S. Paul and Y. Chen, "Spectral and matrix factorization methods for consistent community detection in multi-layer networks," *Ann. Statist.*, vol. 48, no. 1, pp. 230–250, Feb. 2020, doi: 10.1214/18-AOS1800.

[22] R. L. Buckner and L. M. DiNicola, "The brains default network: Updated anatomy, physiology and evolving insights," *Nature Rev. Neurosci.*, vol. 20, pp. 593–608, 2019.

[23] J. Lei, K. Chen, and B. Lynch, "Consistent community detection in multi-layer network data," *Biometrika*, vol. 107, no. 1, pp. 61–73, Dec. 2019, doi: 10.1093/biomet/asz068.

[24] N. Stanley, T. Bonacci, R. Kwitt, M. Niethammer, and P. J. Mucha, "Stochastic block models with multiple continuous attributes," *Appl. Netw. Sci.*, vol. 4, no. 1, Aug. 2019, Art. no. 54, doi: 10.1007/s41109-019-0170-z.

[25] B.-Y. Jing, T. Li, Z. Lyu, and D. Xia, "Community detection on mixture multilayer networks via regularized tensor decomposition," *Ann. Statist.*, vol. 49, no. 6, pp. 3181–3205, 2021, doi: 10.1214/21-AOS2079.

[26] X. Fan, M. Pensky, F. Yu, and T. Zhang, "ALMA: Alternating minimization algorithm for clustering mixture multilayer network," *J. Mach. Learn. Res.*, vol. 23, no. 330, pp. 1–46, 2022. [Online]. Available: http://jmlr.org/papers/v23/21-0182.html

[27] Y. Luo, G. Raskutti, M. Yuan, and A. R. Zhang, "A sharp blockwise tensor perturbation bound for orthogonal iteration," *J. Mach. Learn. Res.*, vol. 22, no. 179, pp. 1–48, 2021. [Online]. Available: http://jmlr.org/papers/v22/20-919.html

[28] A. Zhang and D. Xia, "Tensor SVD: Statistical and computational limits," *IEEE Trans. Inf. Theory*, vol. 64, no. 11, pp. 7311–7338, Nov. 2018.

[29] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Rev.*, vol. 51, no. 3, pp. 455–500, 2009.

[30] T. T. Cai and A. Zhang, "Rate-optimal perturbation bounds for singular subspaces with applications to high-dimensional statistics," *Ann. Statist.*, vol. 46, no. 1, pp. 60–89, 2018, doi: 10.1214/17-AOS1541.

[31] A. Kumar, Y. Sabharwal, and S. Sen, "A simple linear time (1 epsiv;)-approximation algorithm for k-means clustering in any dimensions," in *Proc. 45th Annu. IEEE Symp. Found. Comput. Sci.*, 2004, pp. 454–462.

[32] A. Athreya et al., "Statistical inference on random dot product graphs: A survey," *J. Mach. Learn. Res.*, vol. 18, no. 226, pp. 1–92, 2018. [Online]. Available: http://jmlr.org/papers/v18/17-448.html

[33] C. Gao, Z. Ma, A. Y. Zhang, and H. H. Zhou, "Community detection in degree-corrected block models," *Ann. Statist.*, vol. 46, no. 5, pp. 2153–2185, Oct. 2018.

[34] C. Gao, Z. Ma, A. Y. Zhang, and H. H. Zhou, "Achieving optimal misclassification proportion in stochastic block models," *J. Mach. Learn. Res.*, vol. 18, no. 1, pp. 1980–2024, Jan. 2017.

[35] T. Zhang, A. Szlam, Y. Wang, and G. Lerman, "Hybrid linear modeling via local best-fit flats," *Int. J. Comput. Vis.*, vol. 100, no. 3, pp. 217–240, Dec. 2012, doi: 10.1007/s11263-012-0535-6.

[36] C. M. Le and E. Levina, "Estimating the number of communities in networks by spectral methods," *Electron. J. Statist.*, vol. 16, no. 1, pp. 3315–3342, 2022, doi: 10.1214/21-EJS1971.

[37] D. Donoho, M. Gavish, and E. Romanov, "ScreeNOT: Exact MSE-optimal singular value thresholding in correlated noise," *Ann. Statist.*, vol. 51, no. 1, pp. 122–148, 2023, doi: 10.1214/22-AOS2232.

[38] J. Lei and A. Rinaldo, "Consistency of spectral clustering in stochastic block models," *Ann. Statist.*, vol. 43, no. 1, pp. 215–237, Feb. 2015.

[39] Y. Yu, T. Wang, and R. J. Samworth, "A useful variant of the davis-kahan theorem for statisticians," *Biometrika*, vol. 102, no. 2, pp. 315–323, Apr. 2014, doi: 10.1093/biomet/asv008.

[40] M. D. Domenico, V. Nicosia, A. Arenas, and V. Latora, "Structural reducibility of multilayer networks," *Nature Commun.*, vol. 6, 2015, Art. no. 6864.



**Marianna Pensky** is currently a Professor with the Department of Mathematics, University of Central Florida, Orlando, FL, USA. She is a Fellow of the Institute of Mathematical Statistics and a Fellow of the American Statisical Association. She is an Executive Editor of *Journal of the Statistical Planning and Inference* and an Associate Editor for *Annals of Statistics* and *Journal of the American Statistical Association*.

**Yaxuan Wang** received the M.S. degree in applied mathematics from Capital Normal University, Beijing, China, in 2018, and the Ph.D. degree in mathematics from the University of Central Florida, Orlando, FL, USA, in 2023. She was the recipient of the Graduate Research Excellence Award and the Dean's Fellowship. She has recently started a job as a modeling analyst for risk management in Citi bank, Tampa, Florida.