Journal Pre-proof



Covariate Connectivity Combined Clustering for weighted networks

Zeyu Hu, Wenrui Li, Jun Yan, Panpan Zhang

PII: S0378-4371(26)00923-4
DOI: https://doi.org/10.1016/j.physa.2026.132177
Reference: PHYSA 132177

To appear in: *Physica A*

Please cite this article as: Z. Hu, W. Li, J. Yan et al., Covariate Connectivity Combined Clustering for weighted networks, *Physica A* (2026), doi: https://doi.org/10.1016/j.physa.2026.132177.

This is a PDF of an article that has undergone enhancements after acceptance, such as the addition of a cover page and metadata, and formatting for readability. This version will undergo additional copyediting, typesetting and review before it is published in its final form. As such, this version is no longer the Accepted Manuscript, but it is not yet the definitive Version of Record; we are providing this early version to give early visibility of the article. Please note that Elsevier's sharing policy for the Published Journal Article applies to this version, see: https://www.elsevier.com/about/policies-and-standards/sharing#4-published-journal-article. Please also note that, during the production process, errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

© 2026 Published by Elsevier B.V.

Journal Pre-proof

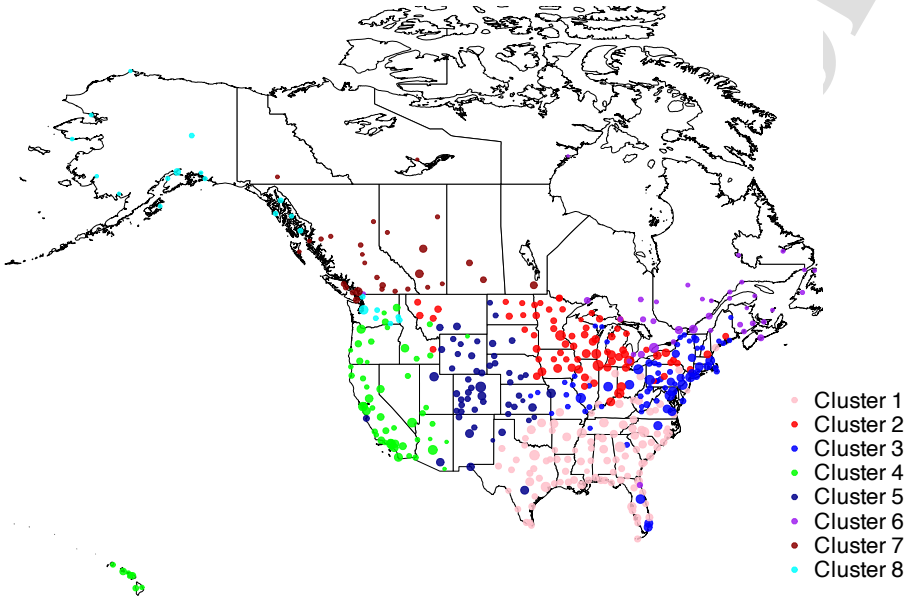
Revised manuscript (marked up)

Click here to view linked References

Graphical Abstract

Covariate Connectivity Combined Clustering for Weighted Networks

Zeyu Hu,Wenrui Li,Jun Yan,Panpan Zhang



Journal Pre-proof

Journal Pre-proof

Highlights

Covariate Connectivity Combined Clustering for Weighted Networks

Zeyu Hu,Wenrui Li,Jun Yan,Panpan Zhang

- Integrates connectivity and node covariates for network clustering.
- Data-driven tuning balances connectivity and covariate signals.
- Spectral clustering with automatic community number selection.

Journal Pre-proof

Journal Pre-proof

Covariate Connectivity Combined Clustering for Weighted Networks

Zeyu Hu[a], Wenrui Li[a], Jun Yan[a] and Panpan Zhang[b,*]

[a]Department of Statistics, University of Connecticut, 215 Glenbrook Road, Unit 4120, Storrs, CT, 06269, USA
[b]Department of Biostatistics, Vanderbilt University Medical Center, 2525 West End Avenue, Suite 1100, Nashville, TN, 37203, USA

ARTICLE INFO

Keywords:
community detection
covariate-assisted learning
network analysis
spectral clustering
similarity fusion
tuning parameter selection

ABSTRACT

Community detection is a central task in network analysis, with applications in social, biological, and technological systems. Traditional algorithms rely primarily on network topology, which can fail when community signals are partly encoded in node-specific attributes. Existing covariate-assisted methods often assume the number of clusters is known, involve computationally intensive inference, or are not designed for weighted networks. We propose C^4: Covariate Connectivity Combined Clustering, an adaptive spectral clustering algorithm that integrates network connectivity and node-level covariates into a unified similarity representation. C^4 balances the two sources of information through a data-driven tuning parameter, estimates the number of communities via an eigengap heuristic, and avoids reliance on costly sampling-based procedures. Simulation studies show that C^4 achieves higher accuracy and robustness than competing approaches across diverse scenarios. Application to an airport reachability network demonstrates the method's scalability, interpretability, and practical utility for real-world weighted networks.

1. Introduction

Community detection is a central problem in network analysis that seeks to identify groups of nodes that are densely interconnected or share similar attributes within a larger network (Girvan and Newman, 2002; Fortunato, 2010). In this paper, we use community detection to refer to the graph clustering task of assigning nodes to latent groups based on network connectivity, node covariates, or both. Uncovering such communities reveals the organizational structures or functional modules that govern complex systems. In social networks, community detection has been widely used to analyze collaboration and interaction patterns (Barabási, Jeong, Néda, Ravasz, Schubert and Vicsek, 2002; Handcock, Raftery and Tantrum, 2007; Ji, Jin, Ke and Li, 2022), with methodological advances further enhancing the modeling of such structures (Ouyang, Dey and Zhang, 2021, 2023). In biological systems, modular organization plays a fundamental role in processes such as brain function (Sporns and Betzel, 2016). In technological and information networks, community detection supports the analysis of incomplete or large-scale structures (Lin, Kong, Yu, Wu, Jia and Li, 2012). Applications also extend to public health, where clustering methods have been employed to group states by disease dynamics and to model covariate-assisted network structures (Chen and Zhang, 2022; Louit, Clark, Gelbard, Vivek, Yan and Zhang, 2025). The ubiquity of network data across these diverse domains underscores the importance of developing reliable community detection methods to uncover hidden structures and gain insights into the organization and function of real-world systems.

Classical community detection methods primarily rely on network topology. Modularity optimization approaches, including the Girvan–Newman algorithm and its successors (Girvan and Newman, 2002; Newman and Girvan, 2004; Newman, 2006), remain influential for their ability to reveal cohesive modules in networks, with the Louvain method especially popular for its efficiency and scalability on large graphs (Blondel, Guillaume, Lambiotte and Lefebvre, 2008). Spectral clustering methods exploit the eigenspectrum of graph Laplacians to identify low-dimensional embeddings of network structure (Shi and Malik, 2000; Ng, Jordan and Weiss, 2002; von Luxburg, 2007), while other graph-based strategies include edge betweenness (Radicchi, Castellano, Cecconi, Loreto and Parisi, 2004), random-walk methods (Pons and Latapy, 2005; Rosvall and Bergstrom, 2008), and label propagation (Raghavan, Albert and Kumara, 2007). Probabilistic formulations such as the mixed-membership stochastic block model extend

*Corresponding author
panpan.zhang@vumc.org (P. Zhang)
ORCID(s): 0000-0003-4401-7296 (J. Yan); 0000-0002-8211-5930 (P. Zhang)

Journal Pre-proof

this topological perspective (Airoldi, Blei, Fienberg and Xing, 2008). These methods effectively detect well-separated clusters but struggle when relevant signals are not fully encoded in connectivity alone, motivating the use of node attributes to enhance community detection.

Covariate-assisted methods extend community detection by combining network topology with node attributes. Covariate-assisted spectral clustering (CASC) integrates connectivity and attributes but requires the number of communities to be specified in advance (Binkiewicz, Vogelstein and Rohe, 2017). To improve performance in sparse networks, regularization strategies have been proposed to stabilize estimation (Yan and Sarkar, 2021). Bayesian frameworks offer complementary strategies. Tallberg (2004) introduced one of the earliest covariate-dependent block models, and Newman and Clauset (2016) formalized inference in annotated networks. Recent developments extend the stochastic block model to incorporate covariates via random partition models (Shen, Amini, Josephs and Lin, 2025), latent factor formulations (Louit et al., 2025), and multilayer settings (Contisciani, Power and De Bacco, 2020; Xu, Zhen and Wang, 2023). Empirical applications also demonstrate the utility of attributes, such as in collaboration networks of statisticians (Zhang, Pan, Wang and Su, 2023). Information-theoretic approaches provide another route for integrating attributes with topology. For instance, Emmons and Mucha (2019) extend the content map equation by introducing a tuning parameter that explicitly trades off metadata agreement against topological description; Peel, Larremore and Clauset (2017) show that node metadata should not be treated as ground-truth communities and propose model-based tools to test and quantify how strongly metadata relate to network structure. To the best of our knowledge, most existing methods address only a subset of the following objectives: supporting weighted connectivity, integrating explicit and interpretable topological-covariate balance, and selecting the number of communities in a data-driven manner.

To fill this gap, we propose Covariate Connectivity Combined Clustering (C^4), an interpretable covariate-assisted spectral clustering framework for weighted networks. The method jointly incorporates weighted network structure and informative node attributes, adaptively selects the tuning parameter that balances these two sources of information, and determines the number of communities in a data-driven manner. In contrast to existing Bayesian formulations, C^4 avoids reliance on computationally intensive Markov chain Monte Carlo procedures and is straightforward to implement. Through extensive simulations, we show that C^4 achieves higher clustering accuracy and robustness than competing methods across a wide range of network configurations, both when the number of communities is known and unknown. Application to an airport reachability network reveals geographically coherent and demographically interpretable communities, further demonstrating its utility and interpretability. Together, these results establish C^4 as a practical and robust solution for covariate-assisted community detection in weighted networks.

The remainder of the paper is organized as follows. Section 2 details the proposed C^4 algorithm, including the construction of the covariate-based similarity matrix, the fusion of connectivity and covariate information, the selection of the tuning parameter, and the practical implementation. Section 3 reports simulation studies conducted under both known and unknown community numbers, evaluating the accuracy and robustness of C^4 across a variety of network configurations. Section 4 presents an application to an airport reachability network, illustrating the scalability and interpretability of the method in practice. Section 5 discusses the limitations and future directions.

2. Method

We begin with introducing notations for weighted networks and covariates (Section 2.1). Then, we detail how connectivity and covariate information are fused within a spectral clustering framework (Section 2.2) and describe the procedure for selecting the tuning parameter (Section 2.3). Implementation details are provided in Section 2.4.

2.1. Notations

We first introduce notations for the weighted network and node-level covariates. Let $G := G(V, E)$ represent a weighted, undirected network with node set V and edge set E. Let $|V| = n$ be the number of nodes in G. The structure of G is characterized by a weighted adjacency matrix $\boldsymbol{W} := (w_{ij})_{n\times n}$, where w_{ij} is the weight between nodes $i, j \in V$. If nodes i and j are not connected, then we set $w_{ij} = 0$. Further, we do not consider self-loops by enforcing $w_{ii} = 0$ for all $i \in V$. For each node i, let $z_i \in \{1, 2, \dots, K\}$ denote its community label, where K is the total number of communities. Collectively, $\boldsymbol{z} = (z_1, z_2, \dots, z_n)^\top$ represents the clustering assignment of all nodes. In practice, K is unknown.

Each node i is endowed with a p-dimensional covariate vector $\boldsymbol{X}_i$ collecting the node-level features. Let $f := \mathbb{R}^p \times \mathbb{R}^p \mapsto \mathbb{R}$ be a similarity function that quantifies the closeness between $\boldsymbol{X}_i$ and $\boldsymbol{X}_j$ for each pair of $i, j \in V$, with larger values indicating greater similarity. Distances followed by inverse transformations are commonly used:

Journal Pre-proof

Euclidean distance for continuous variables, Hamming distance for categorical variables, and more flexible metrics such as Gower's distance for mixed data (Wang, Yabes and Chang, 2021). The choice of inverse transformation depends on both the selected distance function and the practical interpretation of the real data. Let $\boldsymbol{S} := (s_{ij})_{n\times n}$ denote the similarity matrix, with $s_{ij} = f(\boldsymbol{X}_i, \boldsymbol{X}_j)$. In applications, the function f may be chosen to represent either covariate proximity or a more general covariate-derived affinity, depending on the scientific meaning of the nodal attributes.

2.2. Covariate Connectivity Combined Clustering

Our C^4 integrates network topology and node-level covariates within a spectral clustering framework. This method enables uncovering communities that are not only structurally cohesive but also homogeneous with respect to relevant nodal attributes. Define $\boldsymbol{C} := (c_{ij})_{n\times n} = (1-\alpha)\boldsymbol{W} + \alpha\boldsymbol{S}$, where $\alpha \in [0,1]$ is a tuning parameter balancing the structural and covariate similarity. Setting $\alpha = 0$ reduces to structure-only clustering, while $\alpha = 1$ refers to covariate-only clustering. Intermediate values of α fuse the two information sources. Because $\boldsymbol{W}$ and $\boldsymbol{S}$ may differ in scale, we rescale $\boldsymbol{S}$ so that its total sum matches that of $\boldsymbol{W}$:

$$\boldsymbol{S} \leftarrow \boldsymbol{S} \times \frac{\sum_{i,j} w_{ij}}{\sum_{i,j} s_{ij}}. \tag{1}$$

This adjustment ensures that neither source dominates due to magnitude differences.

For any fixed choice of α, spectral clustering is performed on the normalized Laplacian of $\boldsymbol{C}$:

$$\boldsymbol{L}_{\text{norm}} = \boldsymbol{I} - \boldsymbol{D}^{-1/2}\boldsymbol{C}\boldsymbol{D}^{-1/2}, \tag{2}$$

where $\boldsymbol{I}$ is the identity matrix, and $\boldsymbol{D}$ is a diagonal matrix with $d_{ii} := \sum_{j=1}^{n} c_{ij}$. Spectral clustering interprets community detection as an eigenvalue problem, embedding nodes in the eigenspace of $\boldsymbol{L}_{\text{norm}}$ (Chung, 1997; Ng et al., 2002; von Luxburg, 2007). Communities are then identified using k-means applied to the eigenvectors corresponding to the smallest eigenvalues. The number of clusters K is chosen adaptively via an eigengap heuristic (Ng et al., 2002). Specifically, if $\lambda_{(1)}, \lambda_{(2)}, \ldots, \lambda_{(n)}$ are the ordered eigenvalues of $\boldsymbol{L}_{\text{norm}}$, we set

$$K = \arg\max_{k\in\{2,\ldots,n-1\}} \left(\lambda_{(k+1)} - \lambda_{(k)}\right). \tag{3}$$

The multiplicity of the zero eigenvalue of $\boldsymbol{L}_{\text{norm}}$ equals the number of connected components in the graph. In practice, when the graph is connected, there is a single zero eigenvalue; the eigengap heuristic is applied to the ordered sequence of eigenvalues, including the zero(s). This eigengap step is a standard practical rule for selecting K when the number of communities is unknown, and it provides an automated implementation of C^4 rather than a new estimator of K.

2.3. Tuning Parameter Selection

The tuning parameter α is selected through a data-driven grid search that maximizes the silhouette score, a distance-based criterion measuring how well nodes are clustered. Since its appearance (Rousseeuw, 1987), the silhouette score has become a standard internal validation criterion because it simultaneously quantifies within-cluster cohesion and between-cluster separation, while not requiring external labels (Shahapure and Nicholas, 2020). Because of its robustness, direct interpretability, and ease of use, the silhouette score has become a prevalent metric for model selection, particularly in clustering research (Dudek, 2020; Januzaj, Beqiri and Luma, 2023).

To operationalize this selection, the silhouette score is computed from pairwise node distances derived from the fused adjacency-similarity matrix $\boldsymbol{C}$. The distance between nodes i and j is defined as $d(i,j) = 1/(c_{ij} + \varepsilon)$, where c_{ij} denotes the integrated similarity in $\boldsymbol{C}$ and $\varepsilon > 0$ provides numerical stability. This definition ensures that stronger similarities correspond to shorter distances. For each candidate α in the pre-specified grid $\mathbb{A}$, we construct $\boldsymbol{C}$, estimate the number of clusters K by the eigengap heuristic (Section 2.2), and evaluate the silhouette score. The final choice of both α and K is obtained jointly by selecting the pair (α, K) that maximizes the silhouette score. The adaptive aspect of C^4 refers to empirical tuning of the relative contribution of network connectivity and node covariates within the fused representation. The selected value α_{opt} is therefore interpreted as the value favored by the silhouette criterion, rather than as a theoretically optimal estimator of a population-level fusion weight. Establishing oracle optimality of α_{opt} would require a fully specified joint generative model for weighted edges and node covariates alongside a well-defined optimality metric. Thus, the selection of K by the eigengap heuristic and the selection of α by the silhouette-based criterion should be understood as practical procedures for implementing C^4 in data applications, rather than as theoretically guaranteed model-selection methods.

Journal Pre-proof

Algorithm 1: Pseudocode for C^4.

Input: Weighted adjacency matrix $\boldsymbol{W}$; covariate similarity matrix $\boldsymbol{S}$; grid for tuning parameter $\mathbb{A}$; number of clusters K_{true} (optional).

Output: Community labels $\boldsymbol{z}$; optimal tuning parameter α_{opt}; number of clusters K_{opt} (user-provided or data-driven).

Rescale $\boldsymbol{S}$ with Equation (1);
foreach $\alpha \in \mathbb{A}$ **do**
  Construct $\boldsymbol{C} \leftarrow (1-\alpha)\boldsymbol{W} + \alpha\boldsymbol{S}$;
  Compute the normalized Laplacian $\boldsymbol{L}_{\text{norm}}$ of $\boldsymbol{C}$ using Equation (2);
  Calculate and sort the eigenvalues of $\boldsymbol{L}_{\text{norm}}$: $\lambda_{(1)} \le \cdots \le \lambda_{(n)}$;
  if K_{true} is not null **then**
    Set $K(\alpha) \leftarrow K_{\text{true}}$;
  else
    Set $K(\alpha) \leftarrow \arg\max_{k\in\{2,\ldots,n-1\}} \left(\lambda_{(k+1)} - \lambda_{(k)}\right)$;
  end
  Extract community labels $\boldsymbol{z}(\alpha)$ by applying the k-means algorithm to the eigenvectors corresponding to $\lambda_{(1)}, \ldots, \lambda_{(K(\alpha))}$;
  Compute the silhouette score $\text{SilS}(\alpha)$ based on $\boldsymbol{C}$ and $\boldsymbol{z}(\alpha)$ using distance $d(i,j) = 1/(C_{ij} + \varepsilon)$;
end
Set $\alpha_{\text{opt}} \leftarrow \arg\max_{\alpha\in\mathbb{A}} \text{SilS}(\alpha)$, $\boldsymbol{z} \leftarrow \boldsymbol{z}(\alpha_{\text{opt}})$, and $K_{\text{opt}} \leftarrow K(\alpha_{\text{opt}})$;
return $\boldsymbol{z}$, α_{opt}, and K_{opt}.

2.4. Implementation

The pseudocode for C^4 is provided in Algorithm 1. An open-source and user-friendly R package, `c4`, is available on GitHub. The package provides functions for performing spectral clustering, including the proposed C^4 and the covariate-assisted spectral clustering (CASC) method by Binkiewicz et al. (2017), along with additional utilities for preprocessing and evaluation. Although several CASC implementations are publicly available, the original `rCASC` codes from Binkiewicz et al. (2017) did not run properly in our experiment, mainly due to incompatibility with updates of several supporting R package. In addition, the `CASC()` function in the `CASCORE` package (Hu and Wang, 2023) applies only when the covariate dimension exceeds the number of clusters, which limits its applicability. Hence, we developed our own implementation of CASC in package `c4`, with the full workflow illustrated in a vignette, from data generation and preprocessing to clustering. All random seeds used in our simulations are properly documented to ensure reproducibility.

3. Simulations

Simulation study evaluates the performance of the proposed C^4 method under controlled network topology and covariate signal conditions. The study consists of two parts. The first examines scenarios where the number of clusters K is known, while the second considers a more realistic setting where K is unknown. In both parts, C^4 is compared with two fixed baselines: (1) structure-only clustering, which applies spectral clustering to the weighted adjacency matrix $\boldsymbol{W}$ (corresponding to $\alpha = 0$); and (2) covariate-only clustering, which applies spectral clustering to the covariate similarity matrix $\boldsymbol{S}$ (corresponding to $\alpha = 1$). Unlike these baselines, C^4 adaptively selects α from the tuning grid $\mathbb{A}$ to optimize clustering performance. In the simulations, the default grid is $\mathbb{A} = \{0, 0.1, 0.2, \ldots, 1\}$. In practice, users may specify this grid. A finer grid may be used when a more detailed assessment of the connectivity–covariate balance is desired, with additional computational cost. In practice, sensitivity to the grid resolution can be assessed by repeating the analysis with a finer grid and examining the stability of the selected α, K, and resulting community assignments.

Additional benchmark methods are included to compare C^4 with established covariate-assisted, spectral, modularity-based, and model-based community detection approaches. When K is known, we include CASC (Binkiewicz et al., 2017), the Spectral Clustering On Ratios-of-Eigenvectors (Jin, 2015, SCORE), and the stochastic block

Journal Pre-proof

model (Holland, Laskey and Leinhardt, 1983, SBM). When K is unknown, we include Louvain's algorithm (Blondel et al., 2008) and SBM. This design separates the effect of covariate information from the effect of automatic K selection, because CASC requires a specified K, whereas Louvain and SBM can be applied in the unknown-K comparison considered below. All simulation codes and random seeds are provided in the online supplements to ensure reproducibility.

3.1. Data Generation

The simulation design systematically varies network size, cluster number, and covariate and structural signal strengths to evaluate C^4 under conditions of increasing difficulty. Networks of size $n \in \{400, 800\}$ were generated with $K \in \{4, 8\}$ communities. The strength of covariate signal was jointly determined by the distances among cluster means and the noise parameter σ. Specifically, each node was associated with a three-dimensional covariate vector ($p = 3$) drawn from a multivariate normal distribution $\mathrm{MVN}(\boldsymbol{\mu}_k, \sigma^2 \boldsymbol{I})$ if node i belongs to cluster $k \in \{1, 2, \ldots, K\}$, where $\boldsymbol{\mu}_k \in \mathbb{R}^3$ denotes the cluster mean and σ controls within-cluster variability.

The covariate structure controlled overlap among communities through the configuration of cluster centers and the noise parameter σ. When $K = 4$, the cluster centers $\boldsymbol{\mu}_k$ corresponded to the four vertices of a regular tetrahedron centered at the origin with fixed edge length 10. When $K = 8$, they were positioned at the eight vertices of a cube centered at the origin with the same edge length. The degree of overlap, and thus the strength of the covariate signal, was determined jointly by the distances between cluster means and the value of σ. We considered $\sigma \in \{2, 3\}$, representing respectively strong and weak separation. Pairwise dissimilarity between nodes was measured by the Euclidean distance, and the similarity matrix $\boldsymbol{S}$ was defined as $s_{ij} = 1/\|\boldsymbol{X}_i - \boldsymbol{X}_j\|_2$ for $i \neq j$ and $s_{ii} = 0$. This design provided continuous covariates with tunable signal strength for evaluating information fusion in clustering.

On the other hand, the strength of the structural signal was controlled by edge probabilities and weights. Specifically, network topology was generated by fixing the within-community edge probability at $b_{\mathrm{win}} = 0.6$ and varying the between-community probability $b_{\mathrm{btw}} \in \{0.3, 0.4, 0.5\}$. Both unweighted and weighted networks were examined. In the weighted case, each edge weight w_{ij} was independently drawn from $\mathrm{Gamma}(2, \theta)$ with $\theta_{\mathrm{btw}} = 1$ for between-community edges and $\theta_{\mathrm{win}} \in \{1, 1.25, 1.5\}$ for within-community edges. Increasing b_{btw} or decreasing θ_{win} weakened the structural contrast between communities, making the clustering problem more difficult. Together, variation in b_{btw} and θ_{win} provides sensitivity analyses for structural separation and edge-weight distribution. In contrast, larger network size n improved estimation stability and enhanced separation. Each combination of $(n, K, \sigma, b_{\mathrm{btw}})$ and weighting scheme, either unweighted or weighted with a given θ_{win}, was replicated 100 times to ensure reliable comparisons across methods.

3.2. Results

We separately report the results from settings with known K and unknown K.

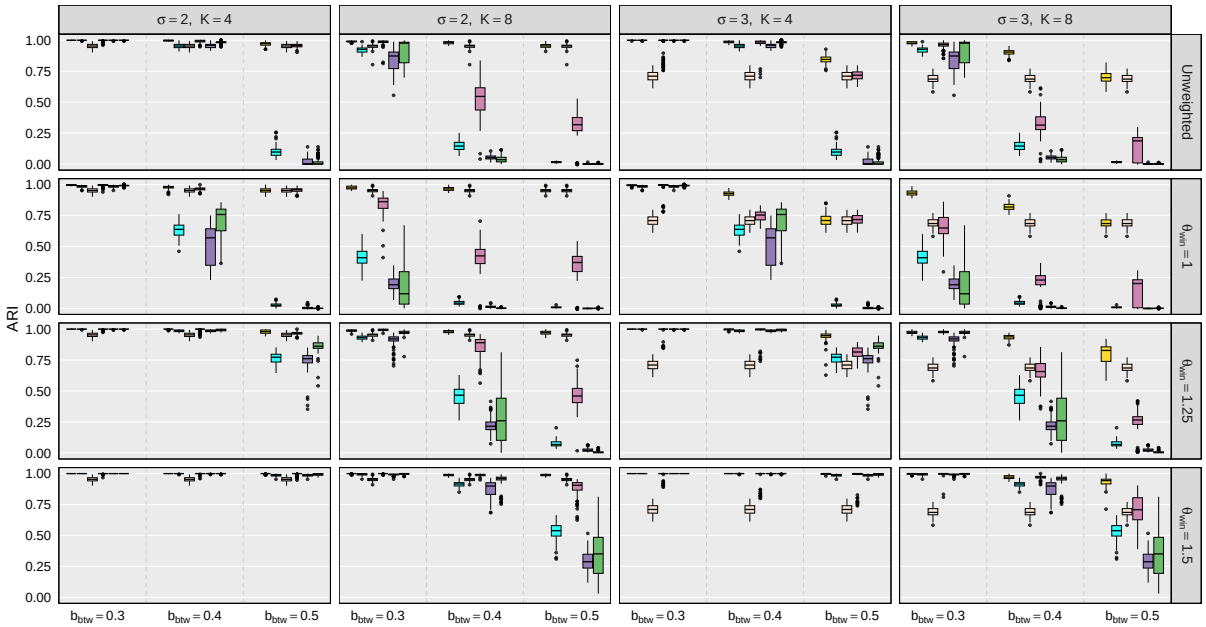
3.2.1. Known K

The experiments with known K evaluated how accurately and consistently C^4 recovered community structures when the true number of clusters was provided. This setting isolates the effect of the fused similarity matrix from errors in estimating K, and it therefore provides a direct comparison of clustering accuracy across methods. Clustering accuracy was measured by the Adjusted Rand Index (ARI, Rand, 1971), which quantifies agreement between estimated and true community assignments, is unaffected by label switching, and remains stable under unbalanced group sizes. Figure 1 displays boxplots of the ARI from 100 replicates under each simulation setting for different methods. In addition to the fixed baselines and CASC, the known-K comparison includes SCORE and SBM as topology-based benchmarks.

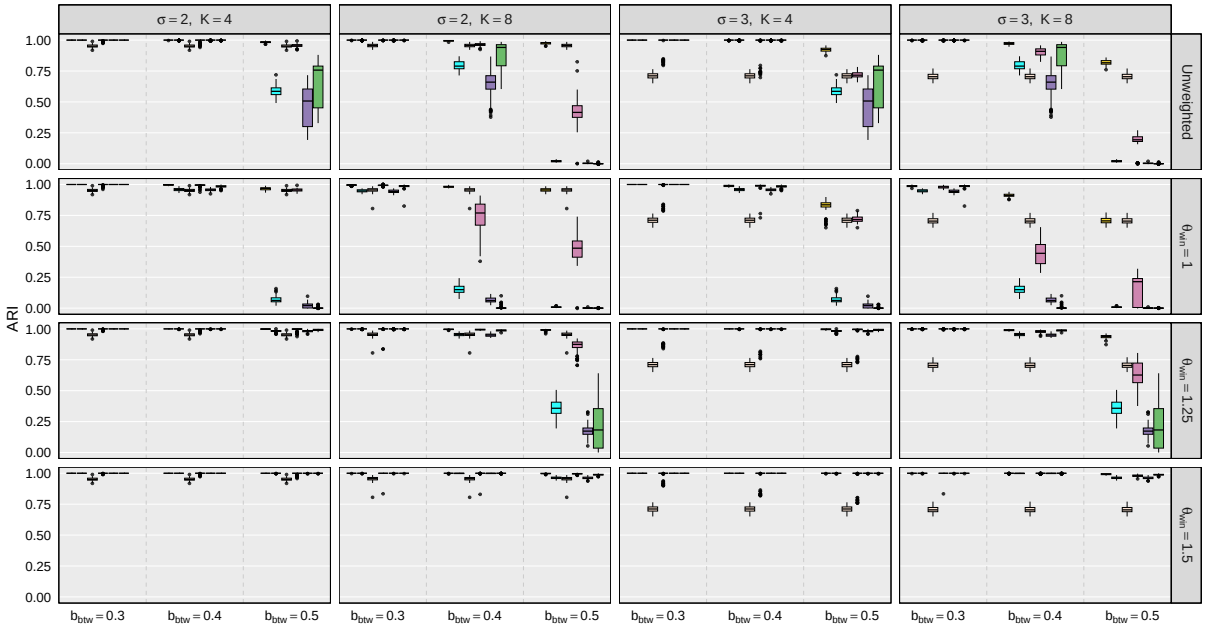
Across all settings, weakening the structural signal lowers accuracy for the baselines, while C^4 remains stable. In unweighted networks, holding σ, n, K and $b_{\mathrm{win}} = 0.6$ fixed, increasing b_{btw} from 0.3 to 0.5 reduces within- and between-community contrast and drives the ARI for $\alpha = 0$ sharply downward in every panel of the top row of Figure 1. CASC tracks C^4 when the structure is strong (e.g., $b_{\mathrm{btw}} = 0.3$) but deteriorates more quickly as b_{btw} grows, with lower medians and greater dispersion. The same pattern appears in weighted networks as θ_{win} decreases. For example, in Figure 1 with $\sigma = 2$, $n = 400$, and $K = 8$, $\alpha = 0$ continues to fall as θ_{win} moves from 1.5 to 1, CASC declines noticeably, and C^4 changes little with small variance. These comparisons imply that CASC is more sensitive to structural weakening, whereas C^4 maintains accuracy and stability. SCORE and SBM follow the same general pattern. They perform well when the network topological structure is informative, but their accuracy declines as the structural signal weakens.

Journal Pre-proof

(a) Network size $n = 400$.



(b) Network size $n = 800$.



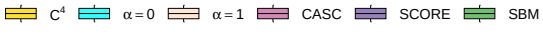


Figure 1: Side-by-side ARI boxplots for known $K \in 4, 8$ with $n = 400$ (upper) and $n = 800$ (lower).

Journal Pre-proof

Table 1
A summary of the selected community number K for $n = 400$, $b_{\text{btw}} = 0.4$, $\theta_{\text{win}} = 1.25$, and true $K = 8$: Sensitivity to the signal-to-noise ratio $\sigma \in \{2, 3\}$. Entries represent frequencies over 100 simulation replications, with the most frequently selected value of K highlighted in blue.

	$\sigma = 2$								$\sigma = 3$							
Method	2	3	4	5	6	7	8	≥ 9	2	3	4	5	6	7	8	≥ 9
$\alpha = 0$	34	20	17	8	11	7	1	2	34	20	17	8	11	7	1	2
$\alpha = 1$	2	0	59	1	0	0	34	4	2	0	95	3	0	0	0	0
C^4	0	0	0	0	0	0	98	2	1	0	56	3	1	7	30	2
Louvain	0	0	0	0	9	45	33	13	0	0	0	0	9	45	33	13
SBM	0	2	8	14	25	19	30	2	0	2	8	14	25	19	30	2

When the covariate signal weakens, $\alpha = 1$ drops, but C^4 remains stable and often outperforms. Comparing Figure 1 at $\sigma = 2$ versus $\sigma = 3$ with $n = 400$ and $K = 8$, both C^4 and CASC achieve high ARI when the structure is strong (e.g., $\theta_{\text{win}} = 1.5$ with $b_{\text{btw}} \in 0.3, 0.4$), indicating limited sensitivity to covariate noise under dominant structural topology. As structural information becomes limited, increasing σ depresses CASC more than C^4: medians shift downward and variability grows across all three b_{btw} levels, while C^4 retains higher medians and smaller dispersion. These patterns show that when covariate signal is weaker and structure is not decisive, C^4 provides a better balance between the two sources and preserves accuracy.

Larger networks and fewer communities improve clustering accuracy for all methods, but C^4 continues to lead in median ARI and stability. Gains are evident as n increases or K decreases across panels, reflecting more stable eigenvectors and easier separation. SCORE and SBM also benefit from larger n and stronger weight contrast, but their performance remains limited when the network structure provides little separation. This limitation is most visible under the extreme setting ($b_{\text{btw}} = 0.5$, $\theta_{\text{win}} = 1$, $K = 8$), where the structure provides little usable information. In this case, C^4 selects $\alpha = 1$ and relies on covariates, yielding reasonable clustering despite negligible topology. Although such cases are uncommon in practice, they illustrate the adaptive mechanism: C^4 shifts weight away from uninformative structure to maintain performance.

3.2.2. Unknown K

The next experiment considered the more realistic setting in which the number of communities K was unknown. For C^4, topology-only clustering ($\alpha = 0$), and covariate-only clustering ($\alpha = 1$), the selection of K followed the procedure described in Algorithm 1. Louvain and SBM determined K through their respective selection procedures.

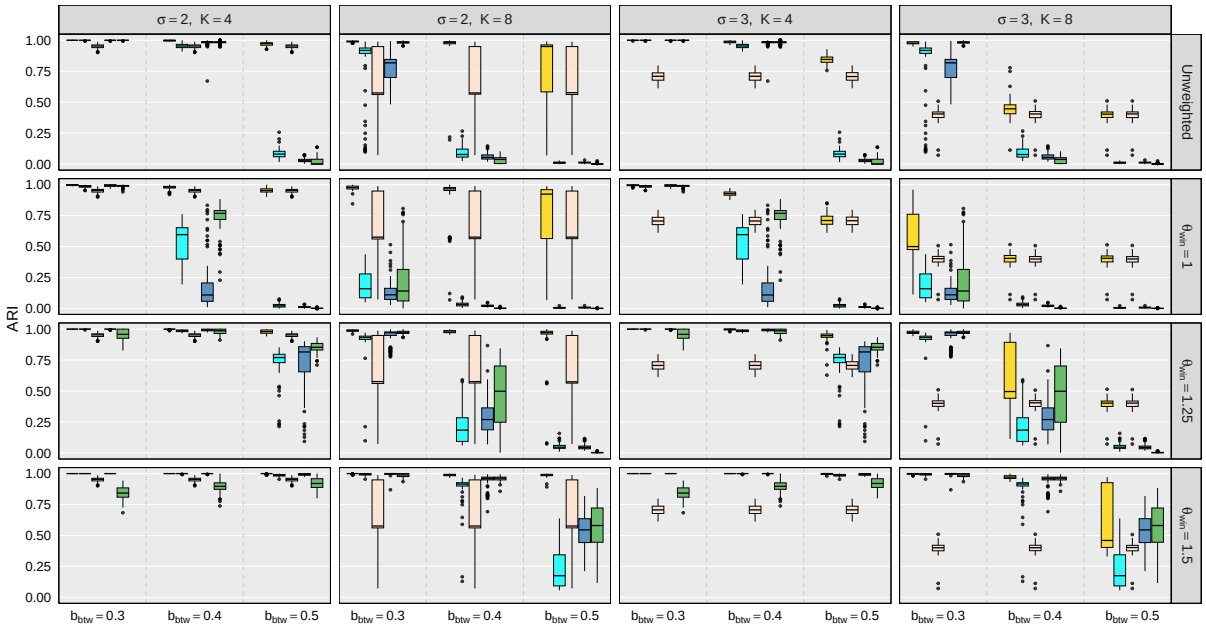
The accuracy of K estimation was examined first. Table 1 presents a representative example with moderate topology structure: $n = 400$, $b_{\text{btw}} = 0.4$, and $K = 8$, using edge weights from a Gamma distribution with scale $\theta_{\text{win}} = 1.25$, a configuration typical of many weighted networks where within-community edges have larger average weights. Results are reported for two levels of the signal-to-noise ratio controlled by $\sigma \in \{2, 3\}$.

The K-selection results show that integrating structure and covariates improves estimation when each source contains partial community information. The baseline method $\alpha = 0$ underestimates the number of clusters, reflecting the difficulty of extracting sufficient community information under moderate network structure. When the covariate signal is strong ($\sigma = 2$), $\alpha = 1$ correctly identifies $K = 8$ in about one-third of cases, but it also tends to underestimate K. By integrating both sources, C^4 achieves near-perfect accuracy and substantially outperforms either baseline. When the covariate signal weakens to $\sigma = 3$, $\alpha = 1$ fails to identify the correct K, whereas C^4 still identifies $K = 8$ in 30% of cases by increasing the weight on structural information. Louvain and SBM show reasonable performance in this representative setting but still tend to underestimate K relative to C^4. The same qualitative pattern appears across other parameter combinations (Appendix A), indicating that C^4 provides more reliable K selection across structural and covariate signal strengths.

Clustering accuracy with unknown K remains highest and most stable for C^4 across all simulation settings. Boxplots in Figure 2 show that, although every method suffers some loss of accuracy relative to the known K case (Figure 1), C^4 consistently attains the highest or tied-for-highest median ARI with smaller dispersion. Louvain and SBM perform well when network communities are strongly separated, but their accuracy decreases as the structural signal weakens. When both structural and covariate information are strong (e.g., $\sigma = 2$, $K = 4$, $n = 400$, $b_{\text{btw}} = 0.3$), C^4 effectively

Journal Pre-proof

(a) Network size $n = 400$.



(b) Network size $n = 800$.

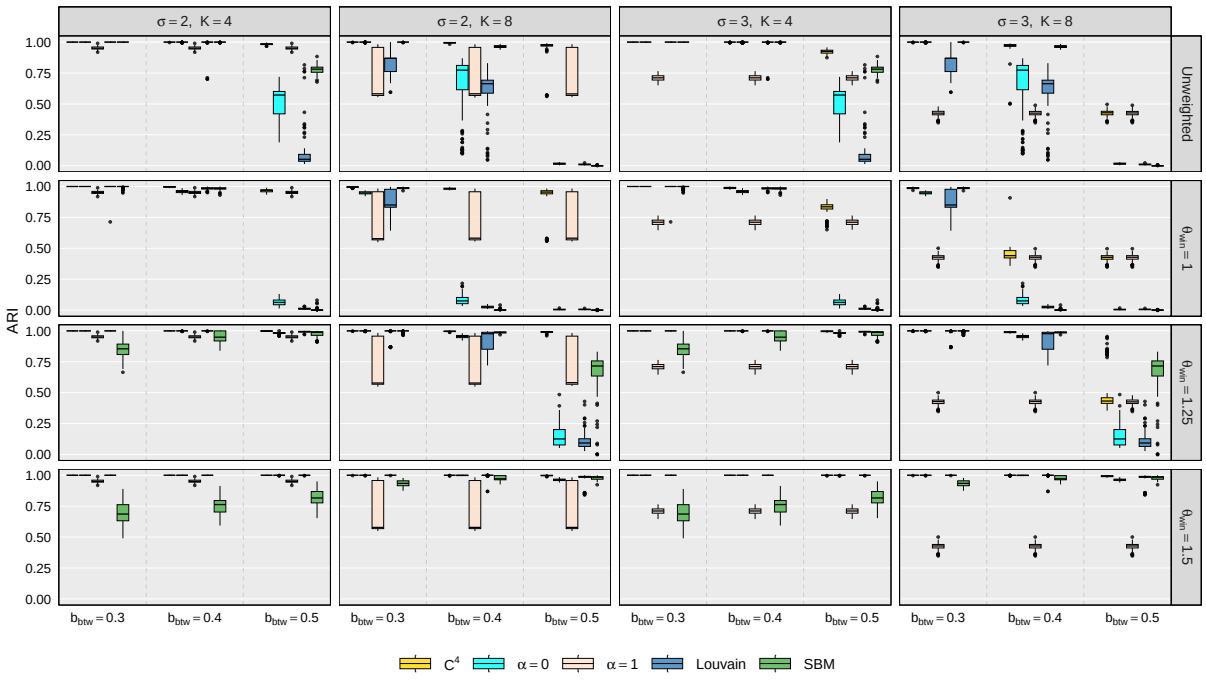


Figure 2: Side-by-side ARI boxplots for unknown $K \in \{4, 8\}$ with $n = 400$ (upper) and $n = 800$ (lower).

Journal Pre-proof

fuses the two sources and matches or surpasses the best competing method performance. When only one source is reliable, either strong topology but weak covariates ($\sigma = 3$) or strong covariates but weak topology ($b_{\text{btw}} = 0.5$), C^4 automatically assigns greater weight to the informative component, maintaining high and stable ARI. Even when both signals are weak, C^4 produces superior accuracy to the competing methods. Patterns remain consistent across weighted networks and different values of n and K. In extreme configurations characterized by weak structure and noisy covariates ($b_{\text{btw}} = 0.5$, $\sigma = 3$, $K = 8$), C^4 tends to select $\alpha = 1$, effectively reducing to covariate-only clustering; details of these cases and the resulting K distributions are provided in Appendix A.

We also conducted an auxiliary simulation with integer-valued Poisson edge weights using the degree-corrected stochastic block model (Karrer and Newman, 2011, DCSBM) as an additional benchmark; details are provided in Appendix B. This analysis was motivated by the importance of degree heterogeneity in many real-world networks. Whereas the main simulations use continuous edge weights, the available DCSBM implementation is designed for integer-valued or count-type edge weights. We therefore generated a separate set of networks with Poisson-distributed edge weights to enable comparison with DCSBM under a compatible setting. In this auxiliary analysis, DCSBM was competitive in some stronger-signal configurations, especially when $K = 4$, but did not provide a uniform advantage when $K = 8$ or when the structural signal was weak. See Appendix B for details.

4. Application to Airline Reachability Network

We applied our method to an Airline Reachability Network (ARN) dataset (Frey and Dueck, 2007). The dataset captures the reachability relationships among 456 major cities in the United States (including Alaska and Hawaii) and Canada, reflecting how easily each city can be accessed by others in terms of estimated commercial airline travel times. Each node represents a city, and a directed edge from city i to city j is established if the estimated airline travel time from i to j, including expected stopover delays, does not exceed 48 hours. The travel times in ARN range from 10 minutes for short regional flights to nearly 48 hours for the longest multi-leg itineraries. Because airline routes are not always reciprocal and prevailing headwinds cause directional differences in flight durations, the ARN exhibits both structural and weight asymmetries: some city pairs are connected only in one direction due to route availability, and even for bidirectional pairs, the edge weights from i to j and from j to i differ because of flight-time asymmetry. Among the 71,959 directed edges, 34,012 city pairs are bidirectional and 3,935 are unidirectional. Among the bidirectional edges, 98% have different weights.

We construct an undirected, weighted network by including only city pairs that are mutually connected via directed edges. The edge weight is computed as the inverse of the mean bidirectional travel time between nodes i and j so that shorter travel times correspond to stronger connections. The resulting undirected ARN exhibits substantial degree heterogeneity: a small number of hub cities connect to hundreds of cities, whereas most smaller regional cities have only a few connections. Such structural characteristics make the ARN a challenging but informative setting for evaluating our C^4 method.

We incorporate a single node-level covariate for each city: the logarithm of the surrounding metropolitan population (Benson, Gleich and Leskovec, 2016). Population size captures the scale of travel demand and the relative importance of a city within the airline mobility system, making it a relevant factor for characterizing node-level heterogeneity. For each city pair, we define a population-based covariate affinity as

$$s_{ij} = \frac{\log(\text{pop}_i) + \log(\text{pop}_j)}{2}.$$

Unlike the distance-based similarity used in the simulations, this quantity is not intended to measure proximity in population size. Instead, it captures the joint population scale, or shared travel-demand potential, associated with a connection between two cities. In other words, the affinity reflects the combined magnitude of the population-based demand signal at the two endpoint cities: two large metropolitan areas receive a high covariate affinity, a pair consisting of one large and one smaller city receives an intermediate affinity, and two smaller cities receive a relatively low affinity. This application-specific definition is appropriate here because population size reflects market scale rather than a clustering feature for which closeness in value is necessarily the desired notion of similarity. More generally, such an affinity is appropriate when a nodal covariate is intended to represent the magnitude or intensity associated with a pair of nodes, rather than homophily or proximity in covariate values.

The ARN application illustrates how C^4 combines travel-time connectivity and population-based affinity in a weighted empirical network. Applying the method gives the selected tuning parameter $\alpha = 0.4$ and selects $K = 8$

Journal Pre-proof

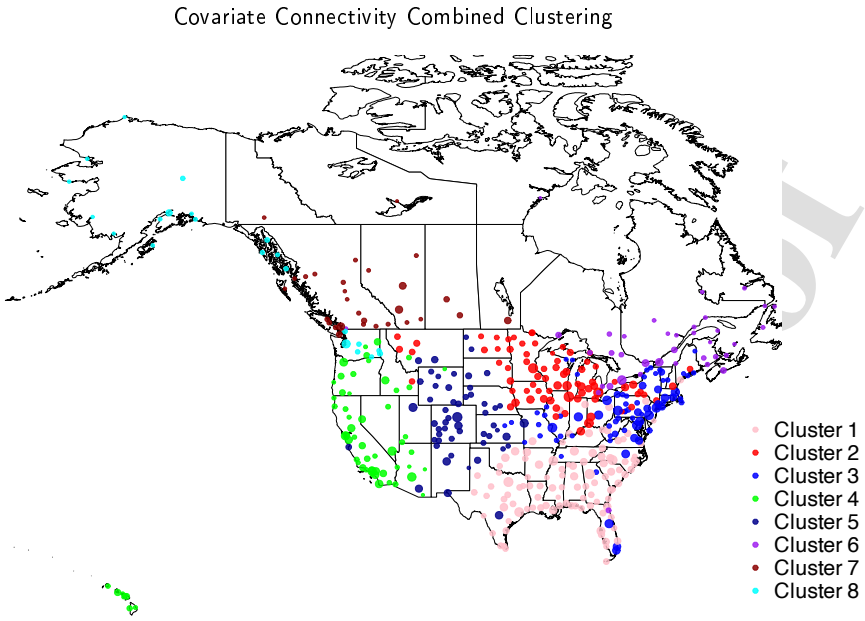


Figure 3: Clustering result based on application of C^4 to the ARN.

Table 2
Summary of community properties of airports across the U.S. and Canada, including community size, largest city (in terms of node strength), and metropolitan population median.

	Size	Largest City	Population Median
Cluster 1	106	Atlanta, GA	357,728
Cluster 2	75	Chicago, IL	187,612
Cluster 3	74	Washington, DC	133,310
Cluster 4	59	Phoenix, AZ	211,888
Cluster 5	54	Denver, CO	46,388
Cluster 6	37	Toronto, ON	53,100
Cluster 7	29	Vancouver, BC	46,850
Cluster 8	22	Seattle, WA	13,732

based on the eigengap heuristic. Figure 3 depicts the community structure, with node size proportional to strength, defined as the sum of incident edge weights. For clarity, edges between nodes are omitted in the figure. Table 2 reports each cluster's size, largest city by node strength, and median metropolitan population, while Table 3 reports within- and between-cluster densities, defined as the proportion of observed edges among all possible node pairs.

The estimated ARN communities show that population scale and connection intensity define distinct dimensions of the airline network. Clusters 1–2 contain large metropolitan areas such as Atlanta and Chicago, combining high population with strong internal ties that form national travel corridors. Clusters 3–4 include major cities such as Washington DC, New York, Phoenix, and Los Angeles, representing large markets with lower densities because of their wide geographic span. Clusters 5, 7, and 8 include metropolitan cities such as Denver, Vancouver, and Seattle that, although smaller than New York or Los Angeles, exhibit high internal cohesion. Cluster 6, with Toronto among its largest members, spans distant Canadian provinces and exhibits the weakest internal linkage. These patterns show that C^4 separates population scale from connectivity intensity in the ARN.

Clusters 1–4 contain populous U.S. regions that anchor the national air-transportation system but differ markedly in their internal connectivity. Cluster 1, centered on Atlanta, the largest city in the group, extends through the Southeast and into the eastern Midwest, while Cluster 2, led by Chicago, links central U.S. markets with the East Coast. In contrast, Cluster 3 comprises several global metropolitan centers including Washington DC, New York, Philadelphia, and Boston, yet extends southward to tourism-linked cities such as Orlando and Miami. Cluster 4, encompassing

Journal Pre-proof

Table 3
Within- and between-community densities (in percentage) based on the clustering result.

	C-1	C-2	C-3	C-4	C-5	C-6	C-7	C-8
C-1	67	47	39	35	29	14	11	11
C-2		68	36	33	33	16	16	14
C-3			47	29	24	22	12	13
C-4				54	45	12	22	36
C-5					71	8	15	15
C-6						45	28	5
C-7							68	14
C-8								71

Phoenix, Los Angeles, Las Vegas, San Francisco, and tourist destinations in Hawaii, comprises major metropolitan areas in the western region and exhibits a hub-dominated structure.

Clusters 5–8 represent smaller but more cohesive regional systems, largely situated in the Mountain West and adjacent western regions of the United States and Canada. Clusters 5, 7, and 8 share a common pattern of compact geography and strong internal ties despite modest population size. Cluster 5, centered on Denver, connects numerous Rocky Mountain cities into a dense network. Cluster 7, dominated by Vancouver, forms a tightly connected western Canadian region, and Cluster 8 links Seattle with airports in Alaska and the Pacific Northwest, making it the smallest yet most internally connected community. In contrast, Cluster 6, with Toronto as its largest city, spans central Canadian provinces, reflecting long distances between major population centers therein. These patterns demonstrate that compact geography, rather than market size, drives cohesion, a distinction the C^4 method captures effectively within the airline network.

5. Discussion

This study proposes a clustering framework, C^4, which adaptively integrates network topology and node-level covariate information through a data-driven weighting mechanism. The method unifies structure- and covariate-based community detection within a single framework that allows both the tuning parameter α and the number of clusters K to be automatically selected or flexibly specified by the user. Because the covariate similarity matrix $\boldsymbol{S}$ is normalized to match the scale of the adjacency matrix $\boldsymbol{W}$, the tuning parameter α gains a clear and interpretable meaning, in contrast to the CASC algorithm (Binkiewicz et al., 2017), where such interpretation is not directly available due to the mismatch of scales. Simulation results show that C^4 consistently achieves accurate and stable recovery of community structures across diverse configurations.

The computational cost of C^4 follows from the spectral decomposition required for each candidate value of α. For each $\alpha \in \mathbb{A}$, the algorithm constructs the fused matrix $\boldsymbol{C}_\alpha$, forms the normalized Laplacian, computes the spectral embedding, and applies k-means. The covariate similarity matrix $\boldsymbol{S}$ is constructed once, with cost $O(n^2 p)$ for p-dimensional covariates. In a dense implementation with full eigen-decomposition, the dominant cost is $O(|\mathbb{A}|n^3)$, in addition to the cost of k-means and silhouette-score evaluation. Thus, C^4 has the same leading-order cost as standard spectral clustering for a fixed α, with an additional multiplicative factor determined by the grid size $|\mathbb{A}|$. Because the covariate similarity matrix is generally dense, sparse matrix acceleration for the original adjacency matrix may not carry over directly to the fused matrix. The main simulation study considers networks of size $n = 400$ and $n = 800$ and is therefore intended primarily to assess clustering performance rather than large-scale scalability. Empirical runtime results for representative simulation settings are provided in Appendix C. Larger-scale computational benchmarking remains an important direction for future work.

Several practical limitations arise from the empirical tuning rule and the user-specified covariate similarity measure. When the covariate information consists of a single categorical variable, the silhouette score may become degenerate, achieving its maximum value of 1 when the clustering is trivially determined by the covariate ($\alpha = 1$). A similar concern may arise when a categorical covariate nearly determines the resulting partition, potentially resulting in values of α close to 1. This degeneracy prevents adaptive selection of α, as the silhouette criterion become uninformative under perfectly separable partitions. Potential remedies include restricting the upper range of α, down-weighting the covariate similarity matrix, or using a penalized selection criterion that discourages purely covariate-defined partitions. The

Journal Pre-proof

current implementation also relies on a fixed grid over α and a chosen similarity measure. Data-driven procedures for selecting the covariate distance or similarity measure, especially for mixed-type covariates, would be a useful extension. Similarly, sensitivity to the covariate representation can be examined by repeating the analysis with scientifically reasonable alternative distance or similarity measures and assessing the stability of the resulting clustering. Further research could also examine how heterogeneous degree distributions or extreme weight variability affect clustering and whether normalization or degree-correction improves robustness. These extensions would clarify how much of the estimated partition is attributable to topology, covariate affinity, or their interaction.

Theoretical guarantees for C^4 remain an important direction for future research. Although α_{opt} is selected through a data-driven criterion in the current implementation, this work does not establish its theoretical optimality. Developing oracle and consistency theory for α under joint weighted-network and covariate generative models is one direction for future work. Existing spectral clustering theory has established consistency under stochastic block models and related settings (von Luxburg, Belkin and Bousquet, 2008; Rohe, Chatterjee and Yu, 2011; Lei and Rinaldo, 2015). However, extending such guarantees to C^4 would require new theoretical development for weighted networks with general covariate similarity matrices and the data-driven selection of α and K. Developing a principled estimator of K within the C^4 formulation is another important direction for future work. Such results would need to specify the population target, the admissible edge-weight and covariate distributions, and the conditions under which the fused Laplacian has a stable eigenspace.

A. Distribution of the Estimated Number of Communities

Figures 4 and 5 present the empirical distributions of the estimated K for $\alpha = 0$, $\alpha = 1$, and C^4, as well as Louvain and SBM, across all simulation configurations. It is worth noting that in extreme scenarios where the structural signal is severely weak and covariate signal is not strong either (e.g. $b_{\text{btw}} = 0.5$, $K = 8$, $\sigma = 3$, and $n \in \{400, 800\}$, with $\theta_{\text{win}} = 1$ or the unweighted network, corresponding to the first and second rows in the last columns of both panels in Figure 5.), the distribution of K selection for C^4 becomes identical to that of $\alpha = 1$. In these scenarios, C^4 automatically places essentially all weight on the covariate information by selecting $\alpha = 1$. Although the $\alpha = 0$ method occasionally estimates the correct $K = 8$ in these extreme settings, cross-comparison with Figure 2 shows that these rare correct identifications still correspond to very poor clustering performance. Therefore, the covariate-only behavior of C^4 in these scenarios represents a rational adaptation that prioritizes clustering quality over nominally correct but uninformative K identification. However, in real applications, such an extreme scenario should prompt careful reassessment of the data and may motivate restriction of the candidate α range based on domain knowledge to obtain more reliable results. The additional topology-based benchmarks show distinct patterns: Louvain tends to recover K more reliably when the true number of communities is smaller, whereas SBM often shows greater variability in selected K, particularly under weaker structural signals and larger K.

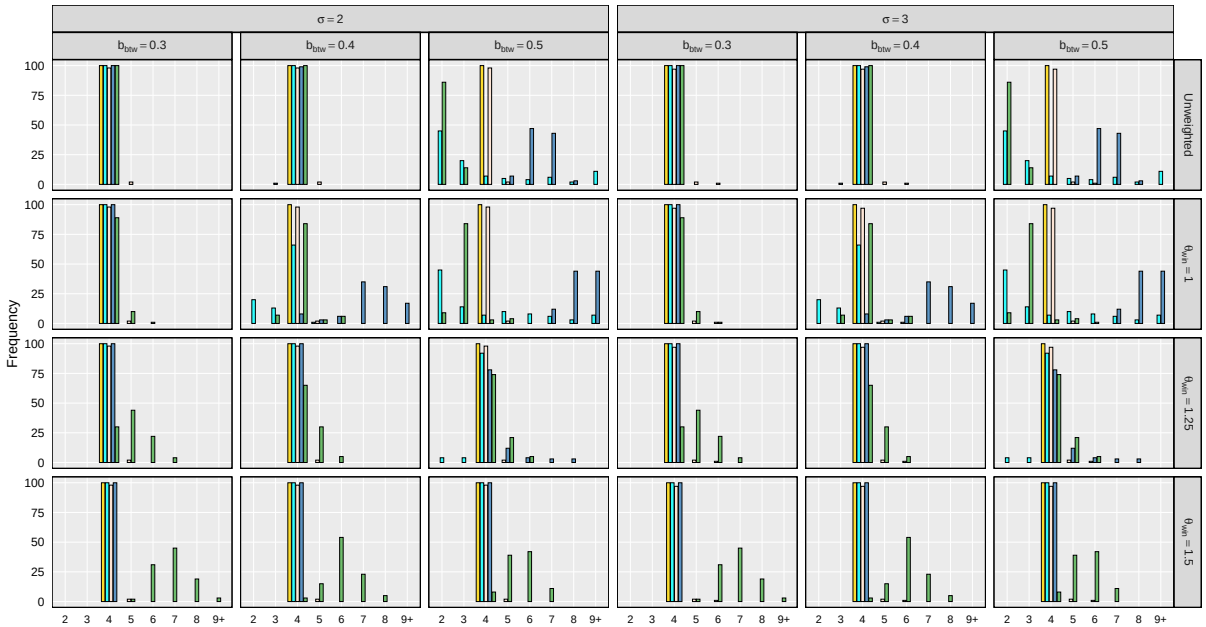
B. Auxiliary Simulation Based on DCSBM

Because the available DCSBM implementation is designed for integer-valued or count-type edge weights, we generated a separate set of weighted networks with positive integer-valued weights. The simulation followed the same general structure as the main simulation study. Networks of size $n \in \{400, 800\}$ were generated with $K \in \{4, 8\}$ communities, and node covariates were generated from cluster-specific multivariate normal distributions with $\sigma \in \{2, 3\}$ controlling covariate separation. Network topology was generated with within-community edge probability fixed at $b_{\text{win}} = 0.6$ and between-community probability $b_{\text{btw}} \in \{0.3, 0.4, 0.5\}$. Conditional on the presence of an edge, integer-valued weights were generated from Poisson distributions, with $\lambda_{\text{btw}} = 1$ for between-community edges and $\lambda_{\text{win}} \in \{1.1, 1.2, 1.3\}$ for within-community edges. Each setting was replicated 100 times. We compared C^4, topology-only clustering ($\alpha = 0$), covariate-only clustering ($\alpha = 1$), Louvain, SBM, and DCSBM, with K unknown.

As shown in Figure 6, C^4 remained the most accurate and stable method across most settings. The advantage was particularly clear when $K = 8$ or when the structural signal weakened, where network-only methods such as Louvain, SBM, and DCSBM deteriorated more rapidly. DCSBM was competitive in some settings with stronger structural information, especially when $K = 4$, but it did not provide a uniform advantage and often performed poorly in the more challenging $K = 8$ or weaker structural configurations. This pattern is consistent with the fact that the auxiliary simulation was designed to make DCSBM applicable to integer-valued weighted networks, but did not impose severe additional node-level degree heterogeneity beyond the block and weight structure.

Journal Pre-proof

(a) Network size $n = 400$.



(b) Network size $n = 800$.

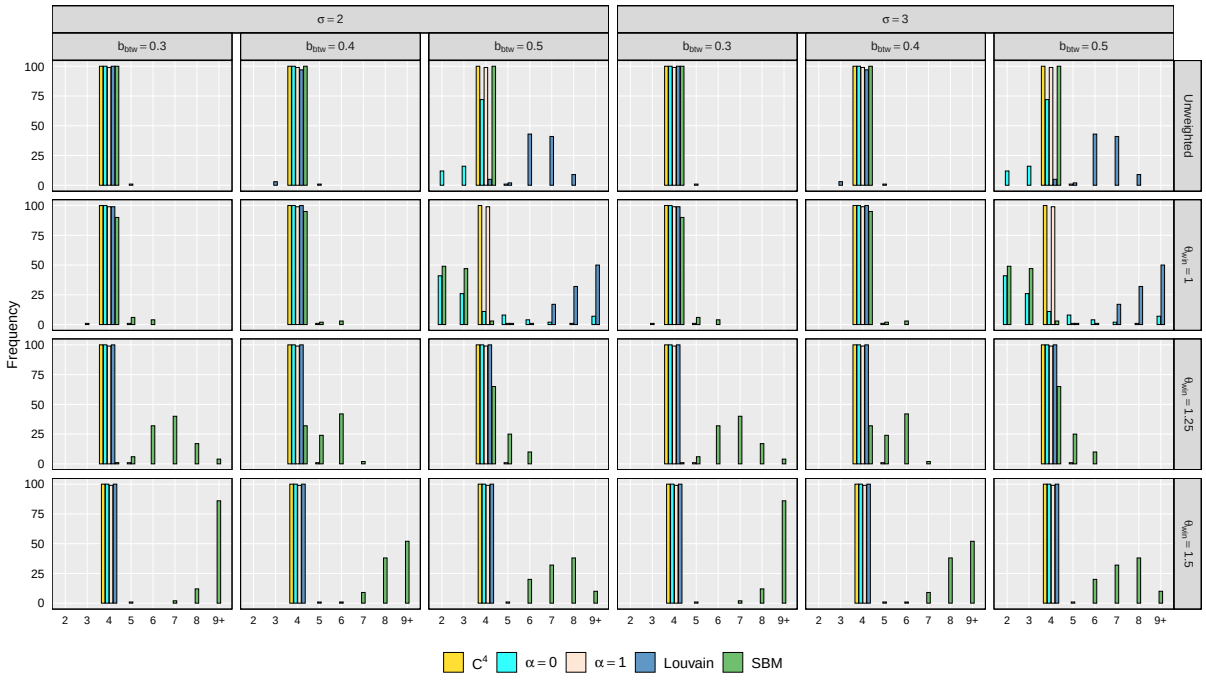
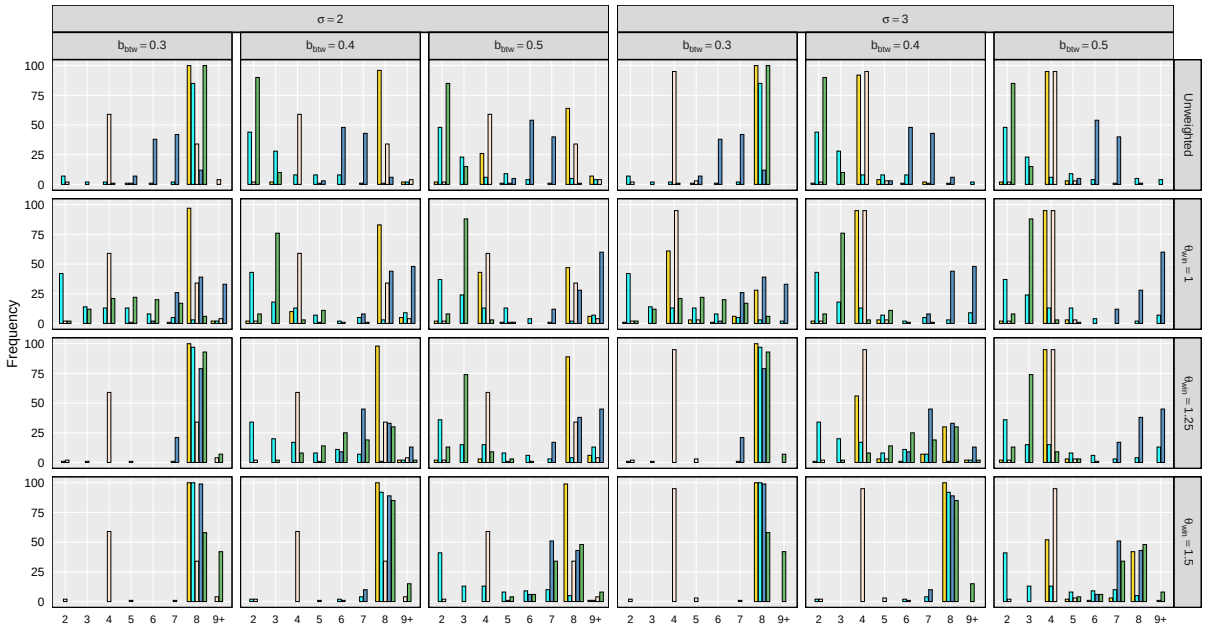


Figure 4: Distribution of the K selection under $K_{\text{true}} = 4$ among all simulation configurations with $n = 400$ (upper) and $n = 800$ (lower).

Journal Pre-proof

(a) Network size $n = 400$.



(b) Network size $n = 800$.

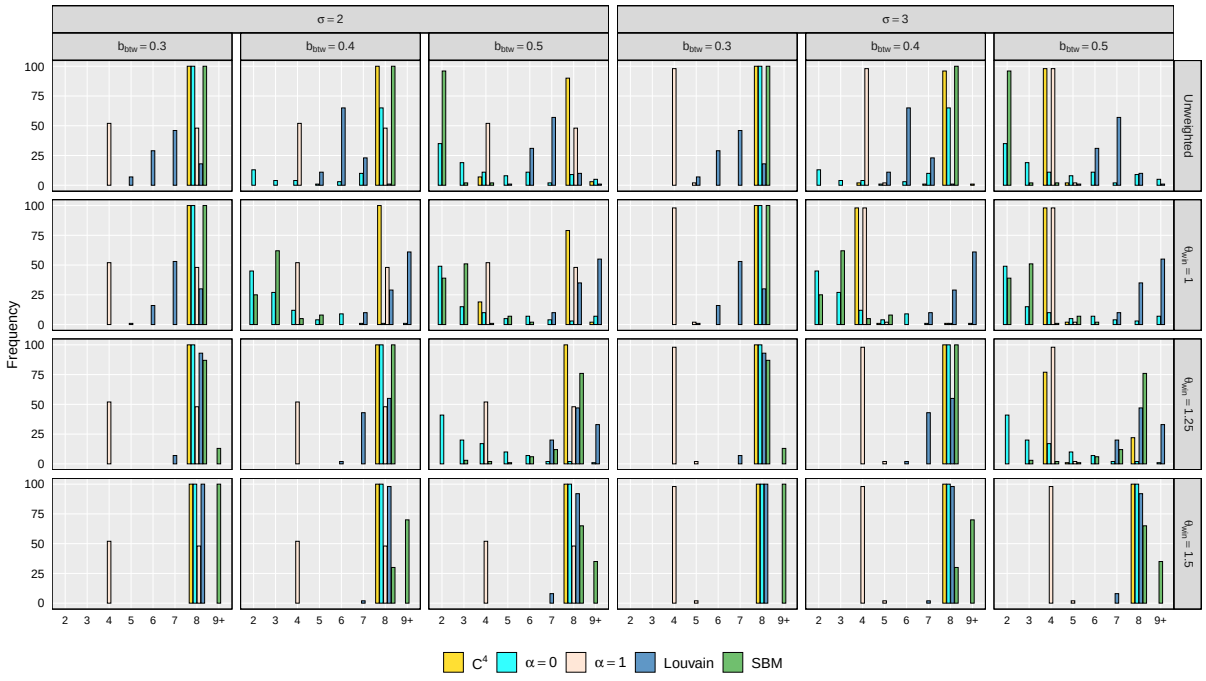


Figure 5: Distribution of the K selection under $K_{\text{true}} = 8$ among all simulation configurations with $n = 400$ (upper) and $n = 800$ (lower).

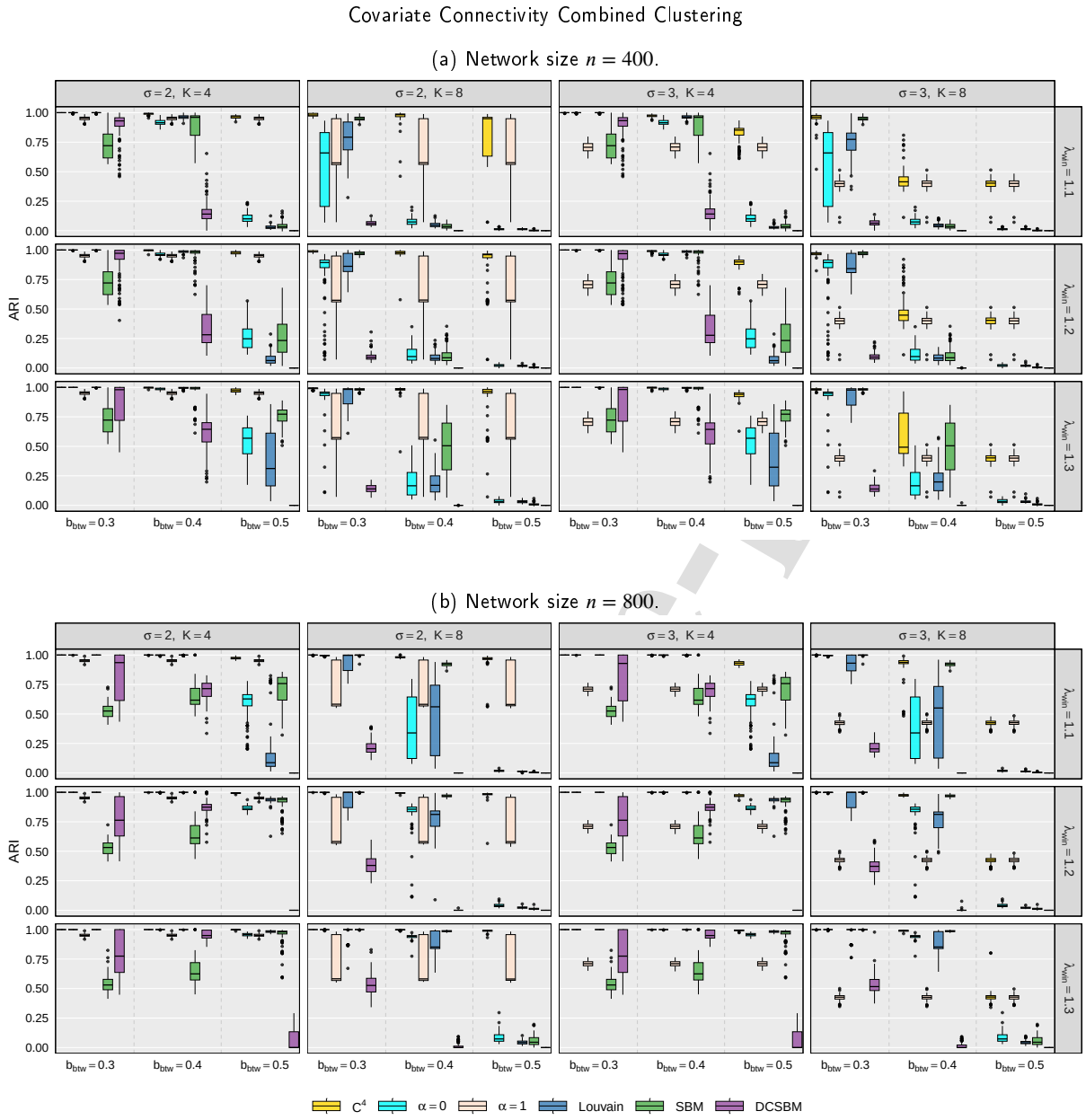


Figure 6: Side-by-side ARI boxplots for unknown $K \in \{4, 8\}$ with $n = 400$ and $n = 800$ under integer-valued Poisson edge weights.

C. Computational Runtime

To provide an empirical assessment of computational cost, we report average algorithm runtime for representative simulation settings. All simulations were conducted in R 4.4.2 on a compute node equipped with two 64-core AMD EPYC 9534 processors, with 101 CPU cores and 220 GB of memory allocated to each array task. The 100 simulation replications were run in parallel. Because runtime varies across network configurations, Table 4 reports the average runtime per replication for representative settings with true $K = 8$, $b_{\text{btw}} = 0.4$, and $\theta_{\text{win}} = 1.25$.

The runtime of C^4 is higher than that of the fixed $\alpha = 0$ and $\alpha = 1$ baselines, as expected because the method evaluates multiple candidate values of α. Across the representative settings considered here, the average runtime of C^4 ranges from approximately 1.9–2.9 seconds for $n = 400$ and 16.5–25.9 seconds for $n = 800$. These results provide

Journal Pre-proof

Table 4
Average algorithm runtime per simulation replication (in seconds) for the representative setting with true $K = 8$, $b_{\text{btw}} = 0.4$, and $\theta_{\text{win}} = 1.25$. The averages are calculated over 100 replications.

K setting	σ	$\alpha = 0$	$\alpha = 1$	C^4	CASC	SCORE	Louvain	SBM
***n*= 400**								
Known K	2	0.17	0.17	1.89	0.64	0.08	–	19.33
	3	0.17	0.18	1.94	0.65	0.08	–	19.21
Unknown K	2	0.23	0.24	2.57	–	–	0.11	511.34
	3	0.26	0.27	2.85	–	–	0.12	493.95
***n*= 800**								
Known K	2	1.38	1.37	16.76	5.35	0.14	–	39.57
	3	1.40	1.33	16.45	5.32	0.14	–	36.57
Unknown K	2	2.31	2.28	25.90	–	–	0.24	2988.74
	3	1.95	1.99	22.13	–	–	0.24	2279.36

an empirical reference for the network sizes studied in this work, rather than a comprehensive large-scale scalability assessment.

CRediT authorship contribution statement

Zeyu Hu: Formal analysis, Investigation, Software, Visualization, Writing - Original Draft. **Wenrui Li:** Supervision, Writing - Review & Editing. **Jun Yan:** Conceptualization, Methodology, Supervision, Writing - Review & Editing. **Panpan Zhang:** Conceptualization, Investigation, Methodology, Software, Supervision, Writing - Original Draft, Writing - Review & Editing.

Acknowledgments

We thank the editor and the four reviewers for their careful reading and constructive suggestions, which have substantially improved the quality of this paper.

References

Airoldi, E.M., Blei, D.M., Fienberg, S.E., Xing, E.P., 2008. Mixed membership stochastic blockmodels. Journal of Machine Learning Research 9, 1981–2014.

Barabási, A.L., Jeong, H., Néda, Z., Ravasz, E., Schubert, A., Vicsek, T., 2002. Evolution of the social network of scientific collaborations. Physica A: Statistical Mechanics and its Applications 311, 590–614.

Benson, A.R., Gleich, D.F., Leskovec, J., 2016. Higher-order organization of complex networks. Science 353, 163–166.

Binkiewicz, N., Vogelstein, J.T., Rohe, K., 2017. Covariate-assisted spectral clustering. Biometrika 104, 361–377.

Blondel, V.D., Guillaume, J.L., Lambiotte, R., Lefebvre, E., 2008. Fast unfolding of communities in large networks. Journal of Statistical Mechanics: Theory and Experiment 2008, P10008.

Chen, J., Zhang, P., 2022. Clustering US states by time series of COVID-19 new case counts in the early months with non-negative matrix factorization. Journal of Data Science 20, 79–94.

Chung, F.R.K., 1997. Spectral Graph Theory. American Mathematical Society, Providence, RI.

Contisciani, M., Power, E.A., De Bacco, C., 2020. Community detection with node attributes in multilayer networks. Scientific Reports 10, 15736.

Dudek, A., 2020. Silhouette index as clustering evaluation tool, in: Classification and Data Analysis (SKAD 2019). Springer, Cham, Switzerland, pp. 19–33.

Emmons, S., Mucha, P.J., 2019. Map equation with metadata: Varying the role of attributes in community detection. Physical Review E 100, 022301.

Fortunato, S., 2010. Community detection in graphs. Physics Reports 486, 75–174.

Frey, B.J., Dueck, D., 2007. Clustering by passing messages between data points. Science 315, 972–976.

Girvan, M., Newman, M.E.J., 2002. Community structure in social and biological networks. Proceedings of the National Academy of Sciences of the United States of America 99, 7821–7826.

Handcock, M.S., Raftery, A.E., Tantrum, J.M., 2007. Model-based clustering for social networks. Journal of the Royal Statistical Society: Series A (Statistics in Society) 170, 301–354.

Holland, P.W., Laskey, K.B., Leinhardt, S., 1983. Stochastic blockmodels: First steps. Social networks 5, 109–137.

Journal Pre-proof

Hu, Y., Wang, W., 2023. CASCORE: Covariate Assisted Spectral Clustering on Ratios of Eigenvectors. Southern Methodist University. R package version 0.1.2.

Januzaj, Y., Beqiri, E., Luma, A., 2023. Determining the optimal number of clusters using silhouette score as a data mining technique. International Journal of Online and Biomedical Engineering 19, 174–182.

Ji, P., Jin, J., Ke, Z.T., Li, W., 2022. Co-citation and co-authorship networks of statisticians. Journal of Business & Economic Statistics 40, 469–485.

Jin, J., 2015. Fast community detection by SCORE. The Annals of Statistics 43, 57–89.

Karrer, B., Newman, M.E., 2011. Stochastic blockmodels and community structure in networks. Physical Review E—Statistical, Nonlinear, and Soft Matter Physics 83, 016107.

Lei, J., Rinaldo, A., 2015. Consistency of spectral clustering in stochastic block models. The Annals of Statistics 43, 215–237.

Lin, W., Kong, X., Yu, P.S., Wu, Q., Jia, Y., Li, C., 2012. Community detection in incomplete information networks, in: Mille, A., Gandon, F., Misselis, J. (Eds.), Proceedings of the 21st International Conference on World Wide Web (WWW'12), Association for Computing Machinery, New York, NY. pp. 341–350.

Louit, S., Clark, E., Gelbard, A., Vivek, N., Yan, J., Zhang, P., 2025. CALF-SBM: A covariate-assisted latent factor stochastic block model. Physica A: Statistical Mechanics and its Applications 667, 130536.

von Luxburg, U., 2007. A tutorial on spectral clustering. Statistics and Computing 17, 395–416.

von Luxburg, U., Belkin, M., Bousquet, O., 2008. Consistency of spectral clustering. The Annals of Statistics 36, 555–586.

Newman, M.E.J., 2006. Modularity and community structure in networks. Proceedings of the National Academy of Sciences of the United States of America 103, 8577–8582.

Newman, M.E.J., Clauset, A., 2016. Structure and inference in annotated networks. Nature Communications 7, 11863.

Newman, M.E.J., Girvan, M., 2004. Finding and evaluating community structure in networks. Physical Review E 69, 026113.

Ng, A.Y., Jordan, M.I., Weiss, Y., 2002. On spectral clustering: Analysis and an algorithm, in: Dietterich, T.G., Becker, S., Ghahramani, Z. (Eds.), Proceedings of the 15th International Conference on Neural Information Processing Systems (NIPS'01), MIT Press, Cambridge, MA. pp. 849–856.

Ouyang, G., Dey, D.K., Zhang, P., 2021. Clique-based method for social network clustering. Journal of Classification 37, 254–274.

Ouyang, G., Dey, D.K., Zhang, P., 2023. A mixed-membership model for social network clustering. Journal of Data Science 21, 508–522.

Peel, L., Larremore, D.B., Clauset, A., 2017. The ground truth about metadata and community detection in networks. Science Advances 3, e1602548.

Pons, P., Latapy, M., 2005. Computing communities in large networks using random walks. Journal of Graph Algorithms and Applications 10, 191–218.

Radicchi, F., Castellano, C., Cecconi, F., Loreto, V., Parisi, D., 2004. Defining and identifying communities in networks. Proceedings of the National Academy of Sciences of the United States of America 101, 2658–2663.

Raghavan, U.N., Albert, R., Kumara, S.T., 2007. Near linear time algorithm to detect community structures in large-scale networks. Physical Review E 76, 036106.

Rand, W.M., 1971. Objective criteria for the evaluation of clustering methods. Journal of the American Statistical Association 66, 846–850.

Rohe, K., Chatterjee, S., Yu, B., 2011. Spectral clustering and the high-dimensional stochastic blockmodel. The Annals of Statistics 39, 1878–1915.

Rosvall, M., Bergstrom, C.T., 2008. Maps of random walks on complex networks reveal community structure. Proceedings of the National Academy of Sciences of the United States of America 105, 1118–1123.

Rousseeuw, P.J., 1987. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics 20, 53–65.

Shahapure, K.R., Nicholas, C., 2020. Cluster quality analysis using silhouette score, in: Webb, G., Zhang, Z., Tseng, V.S., Williams, G., Vlachos, M., Cao, L. (Eds.), Proceedings of IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA), IEEE Conference Publishing Services, Los Alamitos, CA, USA. pp. 747–748.

Shen, L., Amini, A., Josephs, N., Lin, L., 2025. Bayesian community detection for networks with covariates. Bayesian Analysis 20, 735–762.

Shi, J., Malik, J., 2000. Normalized cuts and image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence 22, 888–905.

Sporns, O., Betzel, R.F., 2016. Modular brain networks. Annual Review of Psychology 67, 613–640.

Tallberg, C., 2004. A Bayesian approach to modeling stochastic blockstructures with covariates. Journal of Mathematical Sociology 29, 1–23.

Wang, S., Yabes, J.G., Chang, C.C.H., 2021. Hybrid density- and partition-based clustering algorithm for data with mixed-type variables. Journal of Data Science 19, 15–36.

Xu, S., Zhen, Y., Wang, J., 2023. Covariate-assisted community detection in multi-layer networks. Journal of Business & Economic Statistics 41, 915–926.

Yan, B., Sarkar, P., 2021. Covariate regularized community detection in sparse graphs. Journal of the American Statistical Association 116, 734–745.

Zhang, Y., Pan, R., Wang, H., Su, H., 2023. Community detection in attributed collaboration network for statisticians. Stat 12, e507.

Journal Pre-proof

Journal Pre-proof

Declaration of interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐ The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

Journal Pre-proof