CLUSTERING FOR DIRECTED GRAPHS USING PARAMETRIZED RANDOM WALK DIFFUSION KERNELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Clustering based on the random walk operator has been proven effective for undirected graphs, but its generalization to directed graphs (digraphs) is much more challenging. Although the random walk operator is well-defined for digraphs, in most cases such graphs are not strongly connected, and hence the associated random walks are not irreducible, which is a crucial property for clustering that exists naturally in the undirected setting. To remedy this, the usual workaround is to either naively symmetrize the adjacency matrix or to replace the natural random walk operator by the teleporting random walk operator, but this can lead to the loss of valuable information carried by edge directionality. In this paper, we introduce a new clustering framework, the *Parametrized Random Walk Diffusion Kernel Clustering* (P-RWDKC), which is suitable for handling both directed and undirected graphs. Our framework is based on the diffusion geometry (Coifman & Lafon, 2006) and the generalized spectral clustering framework (Sevi et al., 2022). Accordingly, we propose an algorithm that automatically reveals the cluster structure at a given scale, by considering the random walk dynamics associated with a parametrized kernel operator, and by estimating its critical diffusion time. Experiments on K-NN graphs constructed from real-world datasets and real-world graphs show that our clustering approach performs well in all tested cases, and outperforms existing approaches in most of them.

1 INTRODUCTION

Clustering is a fundamental unsupervised learning task, whose aim is to analyze and reveal the cluster structure of unlabeled datasets, and has widespread applications in machine learning, network analysis, biology, and other fields (Kiselev et al., 2017; McFee & Ellis, 2014). Clustering for data represented as a graph has been formulated in various ways. A well-established one consists in minimizing a functional of the graph-cut (Von Luxburg, 2007; Shi & Malik, 2000), leading to the spectral clustering (SC) framework and various algorithms. SC is simple and effective, but has important limitations (Nadler & Galun, 2006) that make it unreliable in a number of non-rare data regimes. Overcoming these limitations is where the focus of the machine learning community is (Tremblay et al., 2016; Zhang & Rohe, 2018; Dall'Amico et al., 2021; Sevi et al., 2022).

A well-studied clustering approach for high-dimensional data, which is of particular interest for this work, is using the operator associated with a random walk (or, in other terms, with a Markov chain), called *random walk operator*. Meilă & Shi (2001) viewed the pairwise similarities between datapoints as edge flows of a Markov chain and proposed an **ergodic** random walk interpretation of the spectral clustering. However, and beyond spectral clustering, the first to conceive the idea of turning the distance matrix between high-dimensional data into a Markov process were Tishby & Slonim (2000). More specifically, they proposed to examine the decay of mutual information during the relaxation of the Markov process. During the relaxation procedure, the clusters emerge as quasi-stable structures, and then they get extracted using the information bottleneck method (Tishby et al., 2000). Azran & Ghahramani (2006) proposed to estimate the number of data clusters by estimating the diffusion time of the random walk operator that reveals the most significant cluster structure. Lin & Cohen (2010) proposed then to find a low-dimensional embedding of the data using the truncated power iteration of a random walk operator derived from the pairwise similarity matrix.

Clustering high-dimensional data using a random walk operator through the lens of the diffusion geometry (Coifman & Lafon, 2006; Coifman et al., 2005) has also been investigated. Nadler et al. (2006) proposed a unifying probabilistic diffusion framework based on the probabilistic interpretation of spectral clustering and dimensionality reduction algorithms using the eigenvectors of the normalized graph Laplacian. Given the pairwise similarity matrix built from high-dimensional data points, they defined a distance function between any two points based on the random walk on the graph called the diffusion distance (Coifman & Lafon, 2006; Pons & Latapy, 2005) and showed that the low-dimensional representation of the data by the first few eigenvectors of the corresponding random walk operator is optimal under a certain criterion. Recently, an unsupervised clustering methodology has been proposed based on the diffusion geometry framework (Maggioni & Murphy, 2019; Murphy & Polk, 2022), where it has been demonstrated how the diffusion time of the random walk operator can be exploited to successfully cluster datasets for which k-means, spectral clustering, or density-based clustering methods fail.

Although there has been a constant development of clustering approaches which are based on the random walk operator, all such efforts (like those mentioned above) have been proposed for undirected graphs. Moreover, the motivation of most of them is to run forward the random walk to avoid the costly eigendecomposition of the graph Laplacian. The extension of clustering approaches based on the random walk operator to digraphs is much more subtle and challenging. As presented earlier, random walk-based clustering approaches rely either on the eigenvectors of the random walk operator or on its iterated powers. In the directed setting, the random walk is well-defined, but it is not reversible in general like in the undirected case (Levin & Peres, 2017). Consequently, the associated eigenvectors are possibly complex, which makes their interpretation and use difficult in the context of clustering. A possible workaround would be to consider the approach based on the iterated powers of the random walk operator which would allow one to obtain a real-valued embedding and thus avoid the use of complex eigenvectors.

However, a second much more subtle and problematic bottleneck arises: the random walk's irreducibility. While in the undirected case, the random walk is irreducible, i.e. any graph vertex can be reached from any other vertex, this is not the case in general for random walks on digraphs. To overcome the irreducibility issue, either a symmetrization procedure is usually employed in the case where the digraph is derived by a K-NN graph construction or the original random walk operator is replaced by the operator of the *teleporting random walk* (Page et al., 1999). For these two workarounds, valuable information can potentially be discarded. To address this problem, we present a new clustering algorithm based on the diffusion geometry framework, which is suitable for any digraph, either obtained by K-NN constructions or by real digraphs representing asymmetric relationships between vertices (e.g. citation graphs). Our algorithm stems from a new type of graph Laplacians (Sevi et al., 2022) that is parameterized by an arbitrary vertex measure capable of encoding digraph information, and from which we derive a random walk operator. Besides, we can exploit the necessary diffusion time of this parametrized random walk to successfully reveal clusters at different scales.

The contribution of this work is manifold: i) We propose a new similarity kernel operator that we term the ***random walk diffusion kernel***, which derives directly from the diffusion distance. ii) We generalize this kernel through the parametrized random walk operator that we introduced, which allows us to extend the diffusion distance to digraphs. iii) From there, we propose our clustering algorithm on digraphs, the ***parametrized random walk diffusion kernel clustering***, based on the parametrized random walk diffusion kernel operator considered as a data embedding. iv) We propose a method to estimate the diffusion time that reveals at best a given number of clusters, regardless of the graph origin. Finally, we show that our approach is efficient on both synthetic and real-world data as it outperforms existing methods in various cases.

2 BACKGROUND

Graph theory. Let $\mathcal{G} = (\mathcal{V}, \mathcal{E}, w)$ be a (weighted) directed graph (digraph), where $\mathcal{V}$ is the finite set of $N = |\mathcal{V}|$ vertices, and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the finite set of edges. Each edge (i, j) is an ordered vertex pair indicating the direction of a link from vertex i to vertex j. Any function $\nu : \mathcal{V} \to \mathbb{R}_+$, associating a nonnegative value to each graph vertex, can be regarded as a positive vertex measure. The edge weight function $w : \mathcal{V} \times \mathcal{V} \to \mathbb{R}_+$ associates a nonnegative real value to every vertex

pair: $w(i,j) > 0$, iff $(i,j) \in \mathcal{E}$, otherwise $w(i,j) = 0$. A digraph $\mathcal{G}$ is represented by a weighted adjacency matrix $\mathbf{W} = \{w(i,j)\}_{i,j=1}^{N} \in \mathbb{R}_+^{N\times N}$, where $w(i,j)$ is the weight of the edge (i,j). We define the out-degree and the in-degree of the i-th vertex by $d_i^{\text{out}} = \sum_{j=1}^{N} w(i,j)$ and $d_i^{\text{in}} = \sum_{j=1}^{N} w(j,i)$, respectively.

Random walk fundamentals. What we call in short a random walk on a weighted graph $\mathcal{G}$, is defined more formally as a natural random walk on the graph, which is a homogeneous Markov chain $\mathcal{X} = (X_t)_{t\geq 0}$ with a finite state space $\mathcal{V}$, and with state transition probabilities proportional to the edge weights. The entries of the transition matrix $\mathbf{P} = [\,p(i,j)\,]_{i,j\in\mathcal{V}}$ are defined by:

$$p(i,j) = \mathbb{P}(X_{t+1} = j \mid X_t = i) = \frac{w(i,j)}{\sum_{z\in\mathcal{V}} w(i,z)}.$$

Algebraically, the transition matrix $\mathbf{P} \in \mathbb{R}^{N\times N}$ can be expressed as $\mathbf{P} = \mathbf{D}_{\text{out}}^{-1}\mathbf{W}$, $\mathbf{D}_{\text{out}} = \mathbf{D}_{d^{\text{out}}}$ (respectively, $\mathbf{D}_{\text{in}} = \mathbf{D}_{d^{\text{in}}}$), whose spectrum $\text{sp}(\mathbf{P}) \in [-1,1]$. For a strongly connected digraph $\mathcal{G}$, the random walk $\mathcal{X}$ is irreducible. If, in addition, the irreducible random walk $\mathcal{X}$ admits the aperiodicity condition, then $\mathcal{X}$ is ergodic, and therefore as $t \to \infty$, the probability measures, $\forall i \in \mathcal{V}$ and $\delta_i \in \{0,1\}^{N\times 1}$ being the vector output of the Kronecker delta function, converges towards a *unique* stationary distribution denoted by the row vector $\pi \in \mathbb{R}_+^N$ (Brémaud, 2013). Within the undirected setting: $d_i^{\text{out}} = d_i^{\text{in}} = d_i$, where $d \in \mathbb{R}_+^{N\times 1}$ is the vector of the vertex degrees; moreover, the stationary distribution is proportional to the vertex degree distribution, i.e. $\pi \propto d$.

Parametrized graph Laplacians. A notable example of parametrized operators that we use later, is the generalized graph Laplacians from Sevi et al. (2022), which have been proposed as a new type of graph operators for (but not restricted to) digraphs.

Definition 2.1. ***Generalized graph Laplacians*** *(Sevi et al., 2022). Let $\mathbf{P}$ be the transition matrix of a random walk on a digraph $\mathcal{G}$. Under an arbitrary positive vertex measure ν on $\mathcal{G}$, consider the positive vertex measure $\xi = \nu^{\mathsf{T}}\mathbf{P}$. Let also the diagonal matrices $\mathbf{D}_\nu = \text{diag}(\nu)$ and $\mathbf{D}_\xi = \text{diag}(\xi)$. Then, we can define three generalized Laplacians of $\mathcal{G}$ as follows:*

$$\textit{generalized random walk Laplacian}: \quad \mathbf{L}_{\text{RW},(\nu)} = \mathbf{I} - (\mathbf{I} + \mathbf{D}_{\xi/\nu})^{-1}(\mathbf{P} + \mathbf{D}_\nu^{-1}\mathbf{P}^{\mathsf{T}}\mathbf{D}_\nu) \tag{1}$$

$$\textit{unnormalized generalized Laplacian}: \quad \mathbf{L}_{(\nu)} = \mathbf{D}_{\nu+\xi} - (\mathbf{D}_\nu\mathbf{P} + \mathbf{P}^{\mathsf{T}}\mathbf{D}_\nu) \tag{2}$$

$$\textit{normalized generalized Laplacian}: \quad \mathcal{L}_{(\nu)} = \mathbf{D}_{\nu+\xi}^{-1/2}\mathbf{L}_{(\nu)}\mathbf{D}_{\nu+\xi}^{-1/2}. \tag{3}$$

$\mathbf{L}_{\text{RW},(\nu)}$, $\mathbf{L}_{(\nu)}$, and $\mathcal{L}_{(\nu)}$ correspond to the directed graph Laplacians in Chung (2005) when the transition matrix $\mathbf{P}$ is irreducible and the vertex measure ν is the ergodic measure π. These graph Laplacians are parametrized by an arbitrary vertex measure has the modeling capacity to encode the graph directionality using the random walk dynamics of the original digraph.

3 Parametrized random walk operator on graphs

In this section, we introduce the parametrized random walk operator that we propose, and we show how it is derived from the generalized graph Laplacian (presented at the end of Sec. 2).

Definition 3.1. ***Parametrized random walk operator (P-RW).*** *Let $\mathbf{P}$ be the transition matrix of a random walk on a digraph $\mathcal{G}$. Under an arbitrary positive vertex measure ν on $\mathcal{G}$, consider the positive vertex measure $\xi = \nu^{\mathsf{T}}\mathbf{P}$. Let also the diagonal matrices $\mathbf{D}_\nu = \text{diag}(\nu)$ and $\mathbf{D}_\xi = \text{diag}(\xi)$. Finally, let $\mathcal{X}_\nu$ be the random walk on $\mathcal{G}$ with the associated random walk operator (transition matrix) $\mathbf{P}_\nu$ defined by:*

$$\mathbf{P}_{(\nu)} = (\mathbf{I} + \mathbf{D}_{\xi/\nu})^{-1}(\mathbf{P} + \mathbf{D}_\nu^{-1}\mathbf{P}^{\mathsf{T}}\mathbf{D}_\nu). \tag{4}$$

It is easy to check that $\mathbf{P}_{(\nu)}$ is a transition matrix (see Appendix A.1.2). Besides, the parametrized random walk $\mathbf{P}_{(\nu)}$ is directly related to the generalized random walk Laplacian $\mathbf{L}_{\text{RW},(\nu)}$ as shown in the next proposition.

Proposition 3.1. *Let $\mathbf{P}$ be the transition matrix of a random walk on a digraph $\mathcal{G}$. Under an arbitrary positive vertex measure ν on $\mathcal{G}$. The generalized random walk Laplacian and the parametrized random walk on $\mathcal{G}$ are related as follows:*

$$\mathbf{L}_{\text{RW},(\nu)} = \mathbf{I} - \mathbf{P}_{(\nu)}.$$

Proof. We use Eq. 1 of Def. 2.1, and we combine with Eq. 4 of Def. 3.1. □

As stated in Sevi et al. (2022), the generalized random walk Laplacian, $\mathbf{L}_{\text{RW},(\nu)}$, is self-adjoint in $\ell^2(\mathcal{V}, \nu + \xi)$. Consequently, the generalized random walk $\mathbf{P}_{(\nu)}$ is also self-adjoint in $\ell^2(\mathcal{V}, \nu + \xi)$, and the associated random walk $\mathcal{X}_\nu$ is ergodic (under the aperiodicity condition) and admits the ergodic distribution π_ν. Therefore, we have a random walk operator $\mathbf{P}_{(\nu)}$ parameterized by a vertex measure ν that allows the encoding of the random walk dynamic of the original digraph.

4 Diffusion geometry for digraphs

In this section, we review the concept of diffusion geometry (Coifman & Lafon, 2006). We show: i) how its core feature, *the diffusion distance*, can be expressed as a Mahalanobis distance involving a specific kernel matrix, which we call *Random Walk Diffusion Kernel* (RWDK); ii) we show how the SC's diagonalization step can be thought of as a function applied to the spectrum of the Laplacian and the conceptual connection with the RWDK, and finally iii) propose a clustering algorithm based on the generalization of this kernel.

4.1 The Diffusion distance as a Mahalanobis distance

The seminal work by (Coifman & Lafon, 2006) introduced the *diffusion geometry framework*, which uses diffusion processes as basic tool to find meaningful geometric descriptions for dataset. The framework can provide different geometric representations of the dataset by iterating the Markov transition matrix, which is equivalent to running forward the random walk. The key element of the diffusion geometry is the diffusion distance (Coifman & Lafon, 2006; Pons & Latapy, 2005) defined as follows.

Definition 4.1. ***Diffusion distance.*** *Let $\mathbf{P}$ be the transition matrix of a reversible random walk on an undirected graph $\mathcal{G}$, with an ergodic distribution π. The diffusion distance at time $t \in \mathbb{N}$ between the vertices i and j is defined by:*

$$d_t^2(i,j) = \|p_t(i,*) - p_t(j,*)\|_{1/\pi}^2.$$

As noted in (Coifman & Lafon, 2006), the diffusion distance emphasizes the cluster structure, if present. Next, we show that the diffusion distance can be seen as a Mahalanobis distance.

Proposition 4.1. ***Diffusion distance as Mahalanobis distance.*** *Let $\mathbf{P}$ be the transition matrix of a reversible random walk on an undirected graph $\mathcal{G}$, with an ergodic distribution π. $\mathbf{P}$ admits the following eigendecomposition $\mathbf{P} = \mathbf{\Phi}\mathbf{D}_\lambda\mathbf{\Psi}^\mathsf{T}$. The diffusion distance between vertices i and j, $d_t^2(i,j)$ at a given time $t \in \mathbb{N}$ can be expressed as the following Mahalanobis distance:*

$$d_t^2(i,j) = (\delta_i - \delta_j)^\mathsf{T}\mathbf{K}_t(\delta_i - \delta_j),$$

where the similarity positive definite kernel matrix $\mathbf{K}_t$ is defined by:

$$\mathbf{K}_t = \mathbf{P}^{2t}\mathbf{D}_d^{-1}. \tag{5}$$

The diffusion distance as a dissimilarity measure reveals a similarity kernel matrix $\mathbf{K}_t$ that we call ***Random Walk Diffusion Kernel*** **(RWDK)** matrix, which is simply a power of the transition matrix $\mathbf{P}$ normalized by the vertex degrees. Consequently, using the diffusion distance $d_t^2(x,y)$ is equivalent to using the RWDK matrix $\mathbf{K}_t$.

4.2 Random Walk Diffusion Kernel, Normalized Graph Laplacian and Spectral Clustering

Spectral clustering (SC) is one of the most widely used clustering methods due to its simplicity, efficiency, and strong theoretical foundation (Shi & Malik, 2000; Ng et al., 2002; Von Luxburg, 2007; Peng et al., 2015; Boedihardjo et al., 2021). Given a fixed number of clusters k, SC consists of three main steps: i) construct the graph Laplacian matrix; ii) compute the eigenvectors associated with the k smallest eigenvectors of the Laplacian matrix and store them as columns in a matrix; iii) apply k-means on the rows of the latter matrix, which are regarded as embedded representations

Algorithm 1 Parametrized Random Walk Diffusion Kernel Clustering (`P-RWDKC`)

Input: $\mathbf{W} \in \mathbb{R}^{N \times N}$: adjacency matrix, k: number of clusters, ν: vertex measure, t_d: diffusion time
Output: $\boldsymbol{V}_{t_d}$: graph k-partition for diffusion time t_d

1: Compute the parametrized random walk operator $\mathbf{P}_{(\nu)}$, see Eq. 4
2: Compute the parametrized random walk diffusion kernel $\mathbf{K}_{(t_d,\nu)}$, see Eq. 7
3: Consider each $x_i \in \mathbb{R}^N$, $i = 1, ..., N$, to be the embedding of the i-th vertex, represented by the i-th row of $\mathbf{K}_{(t_d,\nu)}$, and apply a clustering method (e.g. k-means) to all these vectors asking for k clusters
4: Obtain the k-partition $\boldsymbol{V}_{t_d} = \left\{V_{t_d}\right\}_{j=1}^{k}$ of the graph vertices based on the clustering result of Step 3
5: **return** $\boldsymbol{V}_{t_d}$

of the datapoints. We aim to highlight how the SC's second step and the RWDK do have similar characteristics. Let us consider the normalized graph Laplacian matrix $\mathcal{L}$ (Chung & Graham, 1997) with eigendecomposition $\mathcal{L} = \sum_{j=1}^{N} \vartheta_j \boldsymbol{\phi}_j \boldsymbol{\phi}_j^{\mathsf{T}}$, ordered eigenvalues $0 \leq \vartheta_1 \leq ... \leq \vartheta_N \leq 2$ and eigenvectors $\{\boldsymbol{\phi}_j\}_{k=1}^{N}$. Computing the eigenvectors associated to the k smallest eigenvalues of $\mathcal{L}$ amounts to applying a function f_1 on the spectrum matrix of $\mathcal{L}$ such that $f_1(x) = 1$ if $x \leq \vartheta_k$, and $f_1(x) = 0$ otherwise, namely $\mathbf{H} = f_1(\mathcal{L}) = \sum_{j=1}^{k} \vartheta_j \boldsymbol{\phi}_j \boldsymbol{\phi}_j^{\mathsf{T}}$. Thanks to the relation $\mathcal{L} = \mathbf{D}^{\frac{1}{2}}(\mathbf{I} - \mathbf{P})\mathbf{D}^{-\frac{1}{2}}$, selecting the k smallest eigenvalues of $\mathcal{L}$ is thus equivalent to selecting the k largest eigenvalues of $\mathbf{P}$, namely $\mathbf{H} = f_1(\mathcal{L}) = \mathbf{D}^{\frac{1}{2}} f_1(\mathbf{I} - \mathbf{P})\mathbf{D}^{-\frac{1}{2}}$. On the other hand, the RWDK $\mathbf{K}_t$ corresponds to applying a function $f_2(x, t) = x^{2t}$ on the spectrum matrix of $\mathbf{P}$. As $t \to \infty$, $f_2(\mathbf{P}, t) \to \pi \mathbf{1}^{\mathsf{T}}$, and consequently $\mathbf{K}_t = f_2(\mathbf{P}, t)\mathbf{D}^{-1} \to [\mathrm{tr}(\mathbf{D}_d)]^{-1}\mathbf{1}\mathbf{1}^{\mathsf{T}}$. As t increases, $f_2(\mathbf{P})$ becomes smoother because f_2 acts as a soft truncation of the spectrum matrix of $\mathbf{P}$, which preserves those eigenvectors of $\mathbf{P}$ that are associated with the largest eigenvalues. However, as t increases, f_1 acts as a hard truncation of the spectrum matrix of $\mathcal{L}$, which preserves the k eigenvectors associated to the lowest ones up to ϑ_k. Consequently, the RWDK $\mathbf{K}_t$ is an adaptive alternative to $\mathbf{H}$, and the aim is to determine the iteration time t that best reveals the k clusters the user looks for.

4.3 Parametrized random walk Kernel clustering

We have shown that the diffusion distance is directly related to the RWDK and that for a given diffusion time t, the RWDK is an alternative to computing the eigenvectors of the graph Laplacian. However, the diffusion distance is only settled for undirected graphs (Coifman & Lafon, 2006; Boyd et al., 2021). Nevertheless, thanks to the parametrized random walk operator built to handle digraphs, one can easily extend the diffusion distance for digraphs without any restriction. This is established by the following definition.

Definition 4.2. ***Parametrized diffusion distance.*** *Let $\mathcal{X}$ be a random walk on digraph $\mathcal{G}$, with transition matrix $\mathbf{P}$. Let ν be an arbitrary positive vertex measure on $\mathcal{G}$, and ξ be the vertex measure defined by $\xi = \nu^{\mathsf{T}}\mathbf{P}$. Define the diagonal matrices $\mathbf{D}_\nu = \mathrm{diag}(\nu)$ and $\mathbf{D}_\xi = \mathrm{diag}(\xi)$. On $\mathcal{G}$, let $\mathcal{X}_\nu$ be a random walk associated with the random walk operator $\mathbf{P}_{(\nu)}$ parametrized by an arbitrary measure ν defined in Eq. 4 and ergodic measure π_ν. Let $p_{t,\nu}(i, *) = \delta_i^{\mathsf{T}}\mathbf{P}_\nu^t$ be the conditional probability vector given the vertex i at the diffusion time t. The parametrized diffusion distance between the vertices i and j, at a given diffusion time t, is defined by:*

$$d_{t,\nu}^2(i, j) = \|p_{t,\nu}(i, *) - p_{t,\nu}(j, *)\|_{1/\pi_\nu}^2 = (\delta_i - \delta_j)^{\mathsf{T}}\mathbf{K}_{t,\nu}(\delta_i - \delta_j), \tag{6}$$

*with $\mathbf{K}_{t,\nu}$ is **the parametrized random walk diffusion kernel** (P-RWDK) defined as:*

$$\mathbf{K}_{t,\nu} = \mathbf{P}_\nu^t \mathbf{D}_{\nu+\xi}^{-1}. \tag{7}$$

Once we have defined the parameterized diffusion distance $d_{t,\nu}^2$, we are in position to propose the novel ***parametrized random walk diffusion kernel clustering*** **(P-RWDKC)** for digraphs, which is based upon the parametrized RWDK $\mathbf{K}_{t,\nu}$ described in Alg. 1. As noted, P-RWDKC has an algorithmic scheme similar to the SC algorithm except for the choice of the operator and the fact that, our method computes the parametrized RWDK $\mathbf{K}_{t,\nu}$ instead of computing the eigenvectors of the graph Laplacian.

5 THE PARAMETRIZED RANDOM WALK DIFFUSION CLUSTERING IN PRACTICE

In the previous section, we described a novel and simple clustering algorithm for digraphs based on an operator parameterized by an arbitrary vertex measure ν and a diffusion time t. To render our algorithm more flexible for practical use, two different aspects have to be considered: i) designing the vertex measure ν, ii) determining the appropriate diffusion time.

5.1 DESIGNING THE VERTEX MEASURE

Designing the vertex measure is one of the major aspects of `P-RWDKC` as we aim to capture with it the random walk dynamics of the original digraph in our parameterized random walk operator. To do so, we propose a vertex measure derived from the iterated powers of a random walk consisting of the forward and backward digraph's flow information. Specifically, the proposed vertex measure can be parametrized by three parameters ($t \in \mathbb{N}$, $\gamma \in [0,1]$, $\alpha \in \mathbb{R}$) and is formally given by:

$$\nu^{\alpha}_{(t,\gamma)}(i) = \left(\tfrac{1}{N}\mathbf{1}^{\mathsf{T}}_{N\times 1}\mathbf{P}^{t}_{\gamma}\delta_i\right)^{\alpha}, \tag{8}$$

where $\mathbf{1}_{N\times 1}$ is the all-ones vector, recall that $\delta_i \in \{0,1\}^{N\times 1}$ is the vector returned by the Kronecker delta function at $i \in \mathcal{V}$, and

$$\mathbf{P}_{\gamma} = \gamma\mathbf{P}_{\text{out}} + (1-\gamma)\mathbf{P}_{\text{in}}, \tag{9}$$

where $\mathbf{P}_{\text{in}} = \mathbf{D}^{-1}_{\text{in}}\mathbf{W}^{\mathsf{T}}$ and $\mathbf{P}_{\text{out}} = \mathbf{D}^{-1}_{\text{out}}\mathbf{W}$ (recall that $\mathbf{D}_{\text{out}} = \mathbf{D}_{d_{\text{out}}}$ and $\mathbf{D}_{\text{in}} = \mathbf{D}_{d_{\text{in}}}$).

The three parameters have an easy-to-see role, and their use is optional as one can set them to values that neutralize their effect (i.e. $t = 1$, $a = 1$, $\gamma = 0.5$). The random walk iteration parameter t controls the random walk diffusion time, γ controls the mixing between $\mathbf{P}_{\text{out}}$ (forward information) and $\mathbf{P}_{\text{in}}$ (backward information), and α controls the re-weighting of the vertex measure. Plugging $\nu^{\alpha}_{(t,\gamma)}$ to Eq. 4 yields the following expression for the parametrized random walk $\mathbf{P}^{(\alpha)}_{t,\gamma} = \mathbf{P}_{(\nu^{\alpha}_{(t,\gamma)})}$ and hence the P-RWDK $\mathbf{K}_{t_d,\nu^{\alpha}_{(t,\gamma)}}$.

5.2 DETERMINING THE APPROPRIATE DIFFUSION TIME IN THE UNSUPERVISED SETTING

For a known number of clusters k, we aim at determining the best diffusion time t_d for our P-RWDK algorithm. As we work in an unsupervised setting, and hence lacking ground truth, determining the right diffusion time is challenging (Shan & Daubechies, 2022). Our main insights to deal with this matter are derived from the concepts of diffusion geometry (Coifman & Lafon, 2006), the theory of nearly uncoupled Markov chains (Tifenbach, 2011; Sharpe & Wales, 2021) and the metastability of Markov chains (Landim & Xu, 2015).

As stated in (Coifman & Lafon, 2006), assuming that the graph has clusters and/or a multi-scale structure, the random walk diffusion reveals clusters at key moments of the diffusion at the course of time. The emergence of clusters can be understood through the prism of metastability theory: for a given cluster structure, there is a critical time-scale when an irreducible random walk becomes nearly reducible, diffusing only inside clusters. Typically, this means to observe approximately that, at that diffusion time, our operator enjoys high intra-cluster compactness and high inter-cluster separation. Cluster validity indexes can be employed for measuring the ratio of the former to the latter of the quantities (José-García & Gómez-Flores, 2021). Here, we propose the use of the Calinski–Harabasz criterion (CH; also known as Variance ratio criterion) (Caliński & Harabasz, 1974), which computes the ratio between the distance of the cluster centroids to the global centroid, and the distance of the datapoints of each cluster to its cluster centroid.

To compute this variance criterion, we propose two types of distances between data points, either the Euclidean distance, or a distance between their embedded representations, which in the context of this work means to consider the rows of a graph operator. Given the set of N datapoints $\boldsymbol{X} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_N\}$ partitioned into k clusters, denoted by $\boldsymbol{V} = \{V_j\}_{j=1}^{k}$, we denote by $\boldsymbol{\mu}_j = \frac{1}{|V_j|}\sum_{\boldsymbol{x}_i \in V_j}\boldsymbol{x}_i$ the centroid of cluster j, by $\boldsymbol{\mu} = \frac{1}{N}\sum_{x_i \in \boldsymbol{X}}\boldsymbol{x}_i$ the centroid of $\boldsymbol{X}$, and by $d(\boldsymbol{x}_i, \boldsymbol{x}_j)$ the pairwise distance used for two datapoints $\boldsymbol{x}_i, \boldsymbol{x}_j \in \boldsymbol{X}$. Thus, the CH criterion en-

Algorithm 2 Estimating the random walk diffusion time t^*

Input: Dataset $\boldsymbol{X}$; $\mathbf{W}$: adjacency matrix; k: number of clusters; J: max number of iterations
Output: t^*: the estimated diffusion time that best reveals k clusters

1: Set $\nu = \mathbf{1}$ for the vertex measure
2: Compute the parametrized random walk operator $\mathbf{P}_{(\nu)}$, see Eq. 4
3: **for** $j = 0$ **to** J **do**
4: Apply k-means on $\mathbf{P}_{(\nu)}^{2^j}$
5: Obtain the k-partition $\boldsymbol{V}_j = \{V_{q,j}\}_{q=1}^k$ of the graph vertices based on the clustering result of Step 4.
6: **end for**
7: Select $j^\star = \underset{j\in\{0,\dots,J\}}{\operatorname{argmax}} \ \mathrm{CH}\big(\boldsymbol{X}, \boldsymbol{V}_j\big)$.
8: **return** $t^* = 2^{j^\star}$

dowed with a given distance d is defined by:

$$\mathrm{CH}\big(\boldsymbol{X}, \boldsymbol{V}\big) = \frac{N-k}{k-1} \frac{\sum_{j=1}^k |V_j| \, d(\boldsymbol{\mu}_j, \boldsymbol{\mu})}{\sum_{j=1}^k \sum_{\boldsymbol{x}_i \in V_j} d(\boldsymbol{x}_i, \boldsymbol{\mu}_j)}.$$

In the standard case where we consider multidimensional data vectors, $\boldsymbol{X} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_N\} \in \mathbb{R}^d$, we can directly use the Euclidean distance $d(\boldsymbol{x}_i, \boldsymbol{x}_j) = \|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2$. However, when we process a graph which is only represented by its adjacency matrix, the CH criterion is no longer directly applicable. For this reason, as part of our framework, we propose to extend the CH criterion for this setting, based on the Kullback–Leibler divergence as a distance Van Erven & Harremos (2014).

Definition 5.1. ***Probability Density-based Calinski-Harabasz (DCH) criterion.*** *Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a digraph with cardinality $|\mathcal{V}| = N$. Let $\mathbf{P}$ be the transition matrix of a random walk on $\mathcal{G}$. The digraph $\mathcal{G}$ is partitioned into k clusters denoted by $\boldsymbol{V} = \{V_j\}_{j=1}^k$. Let $\boldsymbol{p}(i,*) = \delta_i^\mathsf{T}\mathbf{P}$ be the conditional probability vector given the vertex i that we consider as the representation of the vertex i. We define as $\boldsymbol{X} = \big\{\boldsymbol{p}(1,*), ..., \boldsymbol{p}(N,*)\big\} \in \mathbb{R}^N$ the set of conditional probability vectors associated with the vertices of $\mathcal{G}$. Let us denote by $\boldsymbol{\mu}_j = \frac{1}{|V_j|}\sum_{\boldsymbol{p}(i,*)\in V_j} \boldsymbol{p}(i,*)$ the centroid of cluster j, and $\boldsymbol{\mu} = \frac{1}{N}\sum_{\boldsymbol{p}(i,*)\in \boldsymbol{X}} \boldsymbol{p}(i,*)$ the centroid of $\boldsymbol{X}$. The Kullback–Leibler divergence between two discrete probability distributions, $\boldsymbol{p}$ and $\boldsymbol{q}$, is defined as $\mathcal{D}_{\mathrm{KL}}(\boldsymbol{p}, \boldsymbol{q}) = \sum_y p(y) \log \frac{p(y)}{q(y)}$, s.t. $q(y) \neq 0$. Given a set of datapoints $\boldsymbol{X}$ and a partition $\boldsymbol{V}$, the probability density-based Calinski-Harabasz (DCH) criterion endowed with the Kullback–Leibler divergence $\mathcal{D}_{\mathrm{KL}}(\boldsymbol{p}, \boldsymbol{q})$ is defined by:*

$$\mathrm{DCH}\big(\boldsymbol{X}, \boldsymbol{V}\big) = \frac{N-k}{k-1} \frac{\sum_{j=1}^k |V_j| \, \mathcal{D}_{\mathrm{KL}}(\boldsymbol{\mu}_j, \boldsymbol{\mu})}{\sum_{j=1}^k \sum_{\boldsymbol{p}(i,*)\in V_j} \mathcal{D}_{\mathrm{KL}}(\boldsymbol{p}(i,*), \boldsymbol{\mu}_j)}.$$

We thus estimate the diffusion time in practice by evaluating the CH or DCH criterion for the partitions associated with the dyadic powers of the parametric random walk $\mathbf{P}_{(\nu)}^{2^j}$ with $\nu = \mathbf{1}$ (the uniform measure) and $j \in \{0, ..., J\}$ with $J = 15$. Since the purpose is to estimate the diffusion time, taking the vertex measure equal to a uniform measure is sufficient for this situation. We summarize this procedure in Alg. 2.

6 Experiments

6.1 Setup and competitors

In this section, we demonstrate the effectiveness of our approach both on digraphs obtained from high-dimensional data through graph construction procedures, and real-world graphs. When dealing with high-dimensional data, we use the K-nearest neighbor (K-NN) graph construction with $K = \lfloor \log(N) \rfloor$, which produces (relatively) sparse and non-strongly connected digraphs. A resulting K-NN graph is unweighted, directed, and represented by its non-symmetric adjacency matrix $\mathbf{W} = \{w_{ij}\}_{i,j=1}^N$, with entries $w_{ij} = \mathbb{1}\left\{\|x_i - x_j\|^2 \leq \mathrm{dist}_K(x_i)\right\}$. In the latter, $x_i, x_j \in \mathbb{R}^d$ stand for the original coordinates of the datapoints corresponding to the vertices i and j, $\mathrm{dist}_K(x)$ is the Euclidean

distance between x and its K-th nearest neighbor, and $\mathbb{1}\{\,\cdot\,\} \in \{0,1\}$ is the indicator function that evaluates the truth of the input condition.

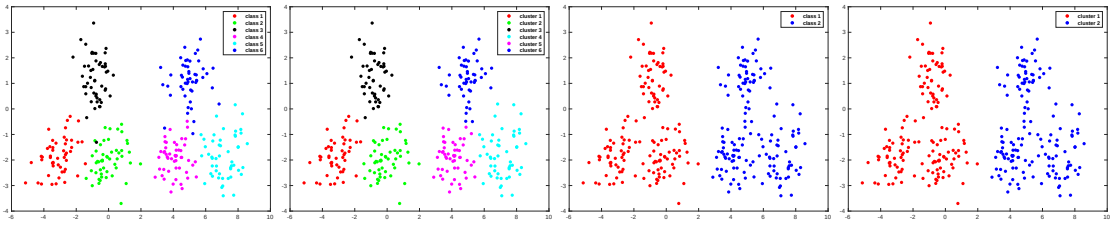
Our clustering method, denoted by $\texttt{P-RWDKC}(\alpha, \gamma, t, t_d)$, is endowed with the parameters $\alpha \geq 0$, $\gamma \in [0,1]$, and $t, t_d \geq 0$. The search grid used for each parameter is: $\alpha \in \{0, 0.1, ..., 1\}$, $t \in \{0, 1, ..., 100\}$, and $\gamma \in \{0, 0.1, ..., 1\}$. Finally, the diffusion time parameter $t_d \in \{2^0, ..., 2^J\}$, with $J = 15$, is estimated using Alg. 2. For each method, we apply k-means clustering over the obtained embeddings (we report the best score out of 100 restarts). We select for each method the optimal parameter values obtained through cross-validation over a grid search, yielding the closest partition to the ground truth. The obtained partitions are evaluated by the normalized mutual information (NMI) Strehl & Ghosh (2002), which is a supervised cluster evaluation index that makes use of the ground truth labels of the data.

In the experiments we compare with the following methods:

- $\texttt{DSC+}(\gamma)$ (Zhou et al., 2005; Palmer & Zheng, 2020) spectral clustering on strongly connected digraphs. To extend this method to the graphs used in our experiments, we employ the teleporting random walk (Page et al., 1999) endowed with the parameter $\gamma \in [0, 1)$.
- $\texttt{DI-SIM}_{\mathrm{L}}(\tau)$ and $\texttt{DI-SIM}_{\mathrm{R}}(\tau)$ (Rohe et al., 2016) are two variants that are based on the left and the right singular vectors, respectively, of a given regularized and normalized operator whose regularization is denoted by the parameter $\tau \geq 0$. We use cross-validation to search the optimal parameter with a grid search over $\tau \in \{1, 2, ..., 20\}$.
- $\texttt{SC-SYM}_1$ and $\texttt{SC-SYM}_2$, two variants of the classical spectral clustering (Von Luxburg, 2007) based on the unnormalized and the normalized graph Laplacians obtained from the symmetrization of the adjacency matrix $\mathbf{W}$.
- $\texttt{PIC}(t_d)$ (Lin & Cohen, 2010) is the clustering approach based on the power iteration of the random walk operator. We use the random walk operator derived from the original adjacency matrix $\mathbf{W}$ of the graph. The diffusion time t_d is estimated using Alg. 2.
- $\texttt{RSC}(\tau)$ (Qin & Rohe, 2013; Zhang & Rohe, 2018) is the regularized spectral clustering proposed to deal with sparse graphs parametrized by $\tau \geq 0$. We use cross-validation with a grid search over $\tau \in \{1, 2, ..., 20\}$ to tune this parameter. The method applies only to undirected graphs. To extend its use to digraphs, we need to symmetrize the adjacency matrix of the original digraph.

6.2 Multi-scale synthetic Gaussians

In this section, we test the effectiveness of our approach at revealing clusters at different scales thanks to the accurate estimation of the diffusion time. We generate *one instance* of a point cloud $X = \{x_i\}_{i=1}^N$, $\forall x_i \in \mathbb{R}^2$, of $N = 300$ data points drawn independently from the following mixture of six Gaussian distributions $\sum_{i=1}^6 \alpha_i \mathcal{N}(\mu_i, \sigma_i^2 \boldsymbol{I})$ with weights $\alpha_i \big(\sum_i \alpha_i = 1\big)$. Specifically, $\sigma_i = 0.5, \alpha_i = 1/6, \forall i$ and $\mu_1 = (-3, -2), \mu_2 = (0, -2), \mu_3 = (-1, 1), \mu_4 = (4, -2), \mu_5 = (7, -2), \mu_6 = (5, 1)$. The resulting data exhibit a multi-scale structure, as they can be seen as either containing the 6 original Gaussian clusters, or as having 2 clusters where each of them is made up of 3 smaller clusters. Figs. 1a and 1c show the ground truth data classes for the two scales. Figs. 1b and 1d, show the respective clustering results obtained by $\texttt{P-RWDKC}$ with a vertex measure $\nu = \mathbf{1}$ and estimated diffusion times $t_{d_1} = 64$ and $t_{d_2} = 128$. The clustering result is quite consistent compared to the ground truth of each case, and hence providing evidence that our framework is able at revealing clusters in a multi-scale fashion.



(a) Ground truth 1 **(b)** 6-means result **(c)** Ground truth 2 **(d)** 2-means result

Figure 1: Comparison between the ground truth classes at different scales and the result of P-RWDKC on a synthetic toy-case. (a),(b) Small scale: ground truth with 6 clusters and the P-RWDKC's result. (c),(d) Large scale: ground truth with 2 clusters and the finding of P-RWDKC.

Table 1: Clustering performance (NMI) on UCI datasets with optimal parameters in brackets.

DATASET	N	d	k	SC-SYM$_1$	SC-SYM$_2$	DI-SIM$_L(\tau)$	DI-SIM$_R(\tau)$	DSC+(γ)	PIC(t_d)	P-RWDKC(α,γ,t,t_d)
IRIS	150	4	3	80.58	80.58	74.98 (1)	66.57 (1)	68.63 (0.80)	78.32 (32)	**90.11 (0.4,1,49,32)**
GLASS	214	9	6	38.59	38.92	38.95 (1)	36.41 (1)	39.72 (0.80)	42.79 (128)	**44.39 (0.9,0.6,1,256)**
WINE	178	13	3	86.33	86.33	83.66 (1)	85.62 (1)	**91.09 (0.80)**	86.33 (4)	86.50 (0.7,0,21,32)
WBDC	569	30	2	67.73	69.47	68.54 (2)	53.43 (1)	61.12 (0.10)	64.77 (8)	**73.90 (0.9,0.3,100,2)**
CONTROL CHART	600	60	6	81.17	81.17	82.94 (1)	77.72 (1)	79.45 (0.90)	82.79 (32)	**87.49 (0.8,1,75,32)**
PARKINSON	185	22	2	21.96	19.13	28.89 (1)	27.36 (13)	25.82 (0.95)	28.89 (2)	**36.08 (1,0.3,52,2)**
VERTEBRAL	310	6	3	39.26	39.26	52.06 (2)	41.76 (2)	56.63 (0.80)	49.13 (8)	**62.40 (0.3,0.8,48,4)**
BREAST TISSUE	106	9	6	54.03	54.43	54.04 (2)	49.33 (2)	51.64 (0.20)	54.18 (32)	**60.43 (0.5,1,39,16)**
SEEDS	210	7	3	73.90	73.90	76.29 (1)	73.06 (1)	74.80 (0.80)	70.79 (32)	**78.95 (1.0,0.4,84,8)**
IMAGE SEG.	2310	19	7	67.06	67.41	67.42 (1)	64.77 (1)	31.83 (0.99)	69.58 (2^{14})	**72.19 (0.8,1,80,256)**
YEAST	1484	8	10	30.58	31.11	31.37 (2)	28.89 (1)	27.50 (0.90)	32.62 (16)	**34.05 (0.7,0.6,51,16)**
AVERAGE	–	–	–	58.29	58.34	59.92	54.77	56.37	60.01	**66.04**

Table 2: Clustering performance (NMI) on real-world datasets with optimal parameters in brackets.

DATASET	N	k	SC-SYM$_1$	RSC(τ)	PIC(t_d)	P-RWDKC(α,γ,t,t_d)
POLITICAL BLOGS (ADAMIC & GLANCE, 2005)	1222	2	1.74	73.25(0.2)	57.37(8)	$\mathbf{75.53}(\mathbf{0.2},\mathbf{1},\mathbf{17},\mathbf{32})$
CORA (BOJCHEVSKI & GÜNNEMANN, 2017)	2485	7	17.10	36.07(2.5)	7.86(4)	$\mathbf{52.03}(\mathbf{0},\mathbf{0.9},\mathbf{72},\mathbf{8})$
COLLEGE FOOTBALL (GIRVAN & NEWMAN, 2002)	115	12	29.97	30.79(2.9)	29.98(2)	$\mathbf{31.30}(\mathbf{0.7},\mathbf{0},\mathbf{17},\mathbf{2})$
KARATE CLUB (ZACHARY, 1977)	34	2	73.24	$\mathbf{83.72}(\mathbf{1.7})$	$\mathbf{83.72}(\mathbf{1})$	$\mathbf{83.72}(\mathbf{0},\mathbf{0},\mathbf{1},\mathbf{1})$
AVERAGE	–	–	30.51	55.96	49.23	**60.65**

6.3 REAL WORLD DATA

In this section, we show that P-RWDKC is consistent with existing methods in clustering high-dimensional data and real networks, and in nearly all tested cases it outperforms them.

High-dimensional data benchmarks. Here we report the results of experiments we conducted to evaluate the performance of the proposed P-RWDKC method on 11 benchmark datasets from the UCI repository (Dheeru & Karra Taniskidou, 2017). We compare against DSC+, SC-SYM$_1$ and SC-SYM$_2$, DI-SIM, and PIC.

Tab. 1 summarizes the comparative results based on NMI. In nearly all cases, the proposed P-RWDKC outperforms significantly the other methods on average. Our approach performs better than SC-SYM$_1$ and SC-SYM$_2$. This allows us to state that P-RWDKC, associated with the suitable vertex measure from Eq. 8 brings indeed real added value to the clustering problem. Furthermore, P-RWDKC outperforms PIC. Consequently, the P-RWDK produces better graph embeddings than the original random walk operator defined the directed K-NN graphs because the parametrized random walk is irreducible compared to the random walk used in PIC that is not and thanks to the digraph information encoded into the vertex measure.

Real-world graph benchmarks. To further validate the efficiency of P-RWDKC, we also evaluated our framework on several real-world graphs. Tab.2 summarizes the comparative results according to the NMI index. Due to space limitation, we include the experiments and explanations in Appendix A.2.

7 CONCLUSION

We have proposed the *parametrized random walk diffusion kernel clustering* (P-RWDKC) that applies to both directed and undirected graphs. First, we introduced the parametrized random walk (P-RW) operator. We then show that the diffusion distance, is a Mahalanobis distance involving a special kernel matrix, called random walk diffusion kernel (RWDK) which is simply a power of the transition matrix (normalized by the vertex degrees). From this, we show that RWDK is an alternative to the eigendecomposition step of the SC pipeline. We propose to extend the diffusion geometry framework to digraphs using combining RWDK and P-RW. Our proposal comes with a clustering algorithm stemming from our framework. Finally, we demonstrated empirically with extensive experiments on several datasets that P-RWDKC outperforms existing approaches for digraphs.

REFERENCES

Lada A Adamic and Natalie Glance. The political blogosphere and the 2004 us election: divided they blog. In *Proceedings of the International Workshop on Link discovery*, pp. 36–43, 2005.

Arik Azran and Zoubin Ghahramani. A new approach to data driven clustering. In *Proceedings of the International Conference on Machine Learning*, pp. 57–64, 2006.

March Boedihardjo, Shaofeng Deng, and Thomas Strohmer. A performance guarantee for spectral clustering. *SIAM Journal on Mathematics of Data Science*, 3(1):369–387, 2021.

Aleksandar Bojchevski and Stephan Günnemann. Deep gaussian embedding of graphs: Unsupervised inductive learning via ranking. *arXiv preprint arXiv:1707.03815*, 2017.

Zachary M Boyd, Nicolas Fraiman, Jeremy Marzuola, Peter J Mucha, Braxton Osting, and Jonathan Weare. A metric on directed graphs and markov chains based on hitting probabilities. *SIAM Journal on Mathematics of Data Science*, 3(2):467–493, 2021.

Pierre Brémaud. *Markov chains: Gibbs fields, Monte Carlo simulation, and queues*, volume 31. Springer Science & Business Media, 2013.

Tadeusz Caliński and Jerzy Harabasz. A dendrite method for cluster analysis. *Communications in Statistics-theory and Methods*, 3(1):1–27, 1974.

Fan Chung. Laplacians and the Cheeger inequality for directed graphs. *Annals of Combinatorics*, 9 (1):1–19, 2005.

Fan RK Chung and Fan Chung Graham. *Spectral graph theory*. Number 92. American Mathematical Soc., 1997.

Ronald R Coifman and Stéphane Lafon. Diffusion maps. *Applied and computational harmonic analysis*, 21(1):5–30, 2006.

Ronald R Coifman, Stephane Lafon, Ann B Lee, Mauro Maggioni, Boaz Nadler, Frederick Warner, and Steven W Zucker. Geometric diffusions as a tool for harmonic analysis and structure definition of data: Diffusion maps. *Proceedings of the National Academy of Sciences*, 102(21): 7426–7431, 2005.

Lorenzo Dall'Amico, Romain Couillet, and Nicolas Tremblay. A unified framework for spectral clustering in sparse graphs. *J. Mach. Learn. Res.*, 22:217–1, 2021.

Lorenzo Dall'Amico, Romain Couillet, and Nicolas Tremblay. Optimal laplacian regularization for sparse spectral community detection. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 3237–3241, 2020.

D Dheeru and E Karra Taniskidou. UCI repository of learnin learning databases. *University of California, Irvine, School of Information and Computer Sciences*, 2017.

Michelle Girvan and Mark EJ Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826, 2002.

Adán José-García and Wilfrido Gómez-Flores. A survey of cluster validity indices for automatic data clustering using differential evolution. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 314–322, 2021.

Vladimir Yu Kiselev, Kristina Kirschner, Michael T Schaub, Tallulah Andrews, Andrew Yiu, Tamir Chandra, Kedar N Natarajan, Wolf Reik, Mauricio Barahona, Anthony R Green, et al. Sc3: consensus clustering of single-cell rna-seq data. *Nature methods*, 14(5):483–486, 2017.

Claudio Landim and Tiecheng Xu. Metastability of finite state markov chains: a recursive procedure to identify slow variables for model reduction. *arXiv preprint arXiv:1512.06597*, 2015.

David A Levin and Yuval Peres. *Markov chains and mixing times*, volume 107. American Mathematical Society, 2017.

Frank Lin and William W Cohen. Power iteration clustering. In *Proceedings of the International Conference on Machine Learning*, 2010.

Mauro Maggioni and James M Murphy. Learning by unsupervised nonlinear diffusion. *J. Mach. Learn. Res.*, 20(160):1–56, 2019.

Brian McFee and Dan Ellis. Analyzing song structure with spectral clustering. In *ISMIR*, pp. 405–410. Citeseer, 2014.

Marina Meilă and Jianbo Shi. A random walks view of spectral segmentation. In *International Workshop on Artificial Intelligence and Statistics*, pp. 203–208, 2001.

James M Murphy and Sam L Polk. A multiscale environment for learning by diffusion. *Applied and Computational Harmonic Analysis*, 57:58–100, 2022.

Boaz Nadler and Meirav Galun. Fundamental limitations of spectral clustering. *Advances in Neural Information Processing Systems*, 2006.

Boaz Nadler, Stephane Lafon, Ronald R. Coifman, and Ioannis G Kevrekidis. Diffusion maps, spectral clustering and eigenfunctions of fokker-planck operators. In *Advances in Neural Information Processing Systems*, 2006.

Andrew Y Ng, Michael I Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems*, 2002.

Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The Pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab, 1999.

William R Palmer and Tian Zheng. Spectral clustering for directed networks. In *International Conference on Complex Networks and Their Applications*, pp. 87–99. Springer, 2020.

Richard Peng, He Sun, and Luca Zanetti. Partitioning well-clustered graphs: Spectral clustering works! In *Conference on Learning Theory*, pp. 1423–1455, 2015.

Pascal Pons and Matthieu Latapy. Computing communities in large networks using random walks. In *International symposium on computer and information sciences*, pp. 284–293. Springer, 2005.

Tai Qin and Karl Rohe. Regularized spectral clustering under the degree-corrected stochastic blockmodel. *preprint arXiv:1309.4111*, 2013.

Karl Rohe, Sourav Chatterjee, and Bin Yu. Spectral clustering and the high-dimensional stochastic blockmodel. *The Annals of Statistics*, 39(4):1878–1915, 2011.

Karl Rohe, Tai Qin, and Bin Yu. Co-clustering directed graphs to discover asymmetries and directional communities. *Proceedings of the National Academy of Sciences*, 113(45):12679–12684, 2016.

Harry Sevi, Matthieu Jonckheere, and Argyris Kalogeratos. Generalized spectral clustering for directed and undirected graphs. *arXiv preprint arXiv:2203.03221*, 2022.

Shan Shan and Ingrid Daubechies. Diffusion maps: Using the semigroup property for parameter tuning. *arXiv preprint arXiv:2203.02867*, 2022.

Daniel J Sharpe and David J Wales. Nearly reducible finite markov chains: Theory and algorithms. *The Journal of Chemical Physics*, 155(14):140901, 2021.

Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

Alexander Strehl and Joydeep Ghosh. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *Journal of Machine Learning Research*, 3(Dec):583–617, 2002.

Ryan M Tifenbach. *A combinatorial approach to nearly uncoupled Markov chains*. PhD thesis, National University of Ireland Maynooth, 2011.

Naftali Tishby and Noam Slonim. Data clustering by Markovian relaxation and the information bottleneck method. *Advances in Neural Information Processing Systems*, 2000.

Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.

Nicolas Tremblay, Gilles Puy, Rémi Gribonval, and Pierre Vandergheynst. Compressive spectral clustering. In *Proceedings of the International Conference on Machine Learning*, pp. 1002–1011, 2016.

Tim Van Erven and Peter Harremos. Rényi divergence and kullback-leibler divergence. *IEEE Transactions on Information Theory*, 60(7):3797–3820, 2014.

Ulrike Von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007.

Wayne W Zachary. An information flow model for conflict and fission in small groups. *Journal of anthropological research*, 33(4):452–473, 1977.

Yilin Zhang and Karl Rohe. Understanding regularized spectral clustering via graph conductance. *preprint arXiv:1806.01468*, 2018.

Dengyong Zhou, Jiayuan Huang, and Bernhard Schölkopf. Learning from labeled and unlabeled data on a directed graph. In *Proceedings of the International Conference on Machine Learning*, pp. 1036–1043, 2005.

A APPENDIX

A.1 PROOFS

A.1.1 PROOF OF PROPOSITION 4.1

Proposition A.1. *Let $\mathcal{X}$ be a random walk on an undirected graph $\mathcal{G}$ with transition matrix $\mathbf{P}$ and ergodic distribution π. The transition matrix $\mathbf{P}$ admits the following eigendcomposition $\mathbf{P} = \mathbf{\Phi}\mathbf{D}_\lambda\mathbf{\Psi}^\mathsf{T}$. The diffusion distance between vertices i and j, $d_t^2(i,j)$ at a given time $t \in \mathbb{N}$ can be written as the following Mahalanobis distance*

$$d_t^2(i,j) = (\delta_i - \delta_j)^\mathsf{T}\mathbf{K}_t(\delta_i - \delta_j),$$

where the similarity positive definite kernel matrix $\mathbf{K}_t$ is defined by

$$\mathbf{K}_t = \mathbf{P}^{2t}\mathbf{D}_d^{-1}. \tag{10}$$

Proof.

$$\begin{aligned} d_t^2(i,j) &= \|p_t(i,*) - p_t(j,*)\|_{1/\pi}^2, \\ &= \|(\mathbf{P}^t)^\mathsf{T}(\delta_i - \delta_j)\|_{1/\pi}^2, \\ d_t^2(i,j) &= (\delta_i - \delta_j)^\mathsf{T}\mathbf{P}^t\mathbf{D}_\pi^{-1}(\mathbf{P}^t)^\mathsf{T}(\delta_i - \delta_j). \end{aligned}$$

By setting $\mathbf{K}_t = \mathbf{P}^t\mathbf{D}_\pi^{-1}(\mathbf{P}^t)^\mathsf{T}$, using the eigendecomposition of $\mathbf{P}$ and the fact that $\mathbf{P}$ is self-adjoint in $\ell^2(\mathcal{V},\pi)$, we have

$$\begin{aligned} \mathbf{K}_t &= \mathbf{P}^t\mathbf{D}_\pi^{-1}(\mathbf{P}^t)^\mathsf{T} \\ &= \mathbf{\Phi}\mathbf{D}_\lambda^t\mathbf{\Psi}^\mathsf{T}\mathbf{D}_\pi^{-1}\mathbf{\Psi}\mathbf{D}_\lambda^t\mathbf{\Phi}^\mathsf{T}, \\ &= \mathbf{\Phi}\mathbf{D}_\lambda^{2t}\mathbf{\Phi}^\mathsf{T}\ (\mathbf{\Psi}^\mathsf{T}\mathbf{D}_\pi^{-1}\mathbf{\Psi} = \mathbf{I}), \\ &= \mathbf{\Phi}\mathbf{D}_\lambda^{2t}\mathbf{\Psi}^\mathsf{T}\mathbf{D}_d^{-1}\ (\mathbf{\Phi} = \mathbf{D}_d^{-1}\mathbf{\Psi}), \\ \mathbf{K}_t &= \mathbf{P}^{2t}\mathbf{D}_d^{-1}. \end{aligned}$$

□

A.1.2 SUPPLEMENTARY PROOFS

Proposition A.2. $\mathbf{P}_{(\nu)}$ *is a transition matrix and reversible.*

Proof. We have the following equality

$$\mathbf{P}_{(\nu)} = (\mathbf{I} + \mathbf{D}_\nu^{-\mathbf{1}}\mathbf{D}_\xi)^{-\mathbf{1}}(\mathbf{P} + \mathbf{D}_\nu^{-\mathbf{1}}\mathbf{P}^\mathsf{T}\mathbf{D}_\nu) = (\mathbf{D}_\nu + \mathbf{D}_\xi)^{-\mathbf{1}}(\mathbf{D}_\nu\mathbf{P} + \mathbf{P}^\mathsf{T}\mathbf{D}_\nu).$$

As a result, we need to show that $(\mathbf{D}_\nu + \mathbf{D}_\xi)^{-\mathbf{1}}(\mathbf{D}_\nu\mathbf{P} + \mathbf{P}^\mathsf{T}\mathbf{D}_\nu)$ is a transition matrix, i.e. we show that

$$\sum_{j=1}^{|\mathcal{V}|}\mathbf{P}_{(\nu),ij} = 1, \quad \forall i \in \mathcal{V}$$

$$\begin{aligned} \sum_{j=1}^{|\mathcal{V}|}\mathbf{P}_{(\nu),ij} &= \sum_{j=1}^{|\mathcal{V}|}\left((\mathbf{D}_\nu + \mathbf{D}_\xi)^{-\mathbf{1}}(\mathbf{D}_\nu\mathbf{P} + \mathbf{P}^\mathsf{T}\mathbf{D}_\nu)\right)_{ij} \\ &= \sum_{j=1}^{|\mathcal{V}|}\left(\sum_{k=1}^{|\mathcal{V}|}(\mathbf{D}_\nu + \mathbf{D}_\xi)_{ik}^{-1}(\mathbf{D}_\nu\mathbf{P} + \mathbf{P}^\mathsf{T}\mathbf{D}_\nu)_{kj}\right) \\ \sum_{j=1}^{|\mathcal{V}|}\mathbf{P}_{(\nu),ij} &= (\mathbf{D}_\nu + \mathbf{D}_\xi)_{ii}^{-1}\sum_{j=1}^{|\mathcal{V}|}(\mathbf{D}_\nu\mathbf{P} + \mathbf{P}^\mathsf{T}\mathbf{D}_\nu)_{ij} \end{aligned} \tag{11}$$

$$\begin{aligned}
\sum_{j=1}^{|\mathcal{V}|}(\mathbf{D}_\nu\mathbf{P}+\mathbf{P}^\mathsf{T}\mathbf{D}_\nu)_{ij} &= \sum_{j=1}^{|\mathcal{V}|}\sum_{k=1}^{|\mathcal{V}|}(\mathbf{D}_\nu)_{ik}\mathbf{P}_{kj}+\sum_{j=1}^{|\mathcal{V}|}\sum_{k=1}^{|\mathcal{V}|}(\mathbf{P}^\mathsf{T})_{ik}(\mathbf{D}_\nu)_{kj} \\
&= \sum_{j=1}^{|\mathcal{V}|}(\mathbf{D}_\nu)_{ii}\mathbf{P}_{ij}+\sum_{j=1}^{|\mathcal{V}|}(\mathbf{P}^\mathsf{T})_{ij}(\mathbf{D}_\nu)_{jj} \\
&= \sum_{j=1}^{|\mathcal{V}|}\nu(i)p(i,j)+\sum_{j=1}^{|\mathcal{V}|}\nu(j)p(j,i) \\
\sum_{j=1}^{|\mathcal{V}|}(\mathbf{D}_\nu\mathbf{P}+\mathbf{P}^\mathsf{T}\mathbf{D}_\nu)_{ij} &= \nu(i)+\xi(i), \quad \forall\, i\in\mathcal{V}
\end{aligned}$$

Using Eq. 11, we finally obtain $\sum_{j=1}^{|\mathcal{V}|}\mathbf{P}_{(\nu),ij}=1$. □

A.2 Experiments on real-world graph benchmarks

Most of the real-world graphs are sparse with heterogeneous degrees with a high variance of the degree. As a result, spectral clustering mostly fails for these graphs Zhang & Rohe (2018); Rohe et al. (2011); Dall'Amico et al. (2020). To avoid numerical issues due to the high variance degree, we proposed a slight modification of the parametrized random walk operator (transition matrix).

$$\mathbf{P}_{(\nu)}=(\mathbf{I}+\mathbf{D}_{\xi/\tilde{\nu}})^{-1}\mathbf{D}_{\text{out}}^{-1}(\mathbf{W}+\mathbf{D}_\nu^{-1}\mathbf{W}^\mathsf{T}\mathbf{D}_\nu). \tag{12}$$

with $\xi=\nu^\mathsf{T}\mathbf{W}$ and the vertex measure $\tilde{\nu}=[\tilde{\nu}_1,...,\tilde{\nu}_N]^\mathsf{T}$ defined by $\tilde{\nu}_i=\nu_i d_i^{\text{out}}, \forall i\in\mathcal{V}$.

We report the results of experiments that evaluate the performance of the proposed `P-RWDKC` method on 4 real-world networks. Among these, 2 are directed (Political blogs, Cora) and 2 are undirected (Karate Club, College Football). For all networks, the number of clusters is considered known. Moreover, when necessary, clustering is performed on the largest connected component of the graph. We compare against $\texttt{SC-SYM}_1$, `RSC`, and `PIC`.

Tab. 2 summarizes the comparative results according to the NMI index. In all cases, the proposed `P-RWDKC` outperforms the other methods, and when looking at the overall average performance the difference is significant. Our approach performs significantly better than $\texttt{SC-SYM}_1$. This is caused by a mixed effect of the symmetrization of the digraph as well as the hard truncation phenomenon discussed in Sec. 4.2. `RSC` is competitive against our method. Nevertheless, the adjacency matrix involved in `RSC` is dense and clearly modified, which can have a deleterious impact compared to our method (e.g. see the results on Cora).

A.3 Alternative for the design of the vertex measure

Designing the vertex measure is one of the major aspects of `P-RWDKC`. We aim to capture the random walk dynamics of the original digraph in our parameterized random walk operator. To do so, we propose a vertex measure derived from the iterated powers of a random walk consisting of the forward and backward digraph's flow information. Specifically, the proposed vertex measure can be parametrized by three parameters ($t\in\mathbb{N}$, $\gamma\in[0,1]$, $\alpha\in\mathbb{R}$) and is formally given by:

$$\nu_{(t,\gamma)}^{\alpha}(i)=\left(\tfrac{1}{N}\mathbf{1}_{N\times 1}^\mathsf{T}\mathbf{P}_\gamma^t\delta_i\right)^\alpha, \tag{13}$$

where $\mathbf{1}_{N\times 1}$ is the all-ones vector , $\delta_i\in\{0,1\}^{N\times 1}$ is the vector output of the Kronecker delta function at $i\in\mathcal{V}$, and

$$\mathbf{P}_\gamma=\mathbf{D}_\gamma^{-1}\mathbf{W}_\gamma, \quad \mathbf{D}_\gamma=\text{diag}(\mathbf{W}_\gamma\mathbf{1}), \tag{14}$$

where $\mathbf{W}_\gamma$ is defined by

$$\mathbf{W}_\gamma=\gamma\mathbf{W}+(1-\gamma)\mathbf{W}^\mathsf{T}, \quad \gamma\in[0,1]. \tag{15}$$

The random walk iteration parameter t controls the random walk diffusion, γ controls the influence between the original adjacency $\mathbf{W}$ (forward information) and its transpose $\mathbf{W}^\mathsf{T}$ (backward information), and α controls the re-weighting of the vertex measure.

A.3.1 VERTEX MEASURE WHEN $\gamma = 1/2$ AND $t \to \infty$

In the setting where, $\gamma = \frac{1}{2}$ and $t \to \infty$, we are able to characterize explicitly the vertex measure. Indeed, $\mathbf{W}_\gamma$ becomes symmetric, the associated random walk is thus ergodic, $\lim_{t\to\infty} \delta_i^\intercal \mathbf{P}_{1/2}^t = \pi_{1/2}, \forall i$ and the parametrized vertex measure defined in Eq. 13 becomes $\nu^\alpha_{(t,\gamma)}(i) = \big(\pi_{1/2}(i)\big)^\alpha$ for any vertex i. Consequently, in this setting, the vertex measure is only parametrized by α.

A.3.2 ADDITIONAL EXPERIMENTS

In this section, we demonstrate the effectiveness of our approach based on the same setting described in Section 6, using the vertex measure in Section A.3.1. Tabs. 3 and 5 summarize the comparative results based on NMI. Similarly to the results from Sections 6.3 and 6.3, we observe that the proposed P−RWDKC outperforms significantly the other methods in nearly all cases. The P−RWDKC, associated with the vertex measure defined in Section A.3.1 stays competitive against P−RWDKC from Section 6.3 with less degree of freedom. The main takeway to consider is that the construction of this kernel and the design of the vertex measures are the key elements to creating relevant embedding for clustering graphs.

Table 3: Clustering performance (NMI) on UCI datasets with optimal parameters in brackets.

DATASET	N	d	k	SC−SYM$_1$	SC−SYM$_2$	DI−SIM$_\mathrm{L}(\tau)$	DI−SIM$_\mathrm{R}(\tau)$	DSC+(γ)	PIC(t_d)	P−RWDK(α, t_d)
IRIS	150	4	3	80.58	80.58	74.98(1)	66.57(1)	68.63(0.80)	78.32(32)	**90.11**$(\mathbf{0.6}, \mathbf{4})$
GLASS	214	9	6	38.59	38.92	38.95 (1)	36.41 (1)	39.72(0.80)	42.79(128)	**42.89**$(\mathbf{0.5}, \mathbf{256})$
WINE	178	13	3	86.33	86.33	83.66(1)	85.62(1)	**91.09**(0.80)	86.33(4)	84.73$(0, 32)$
WBDC	569	30	2	67.73	69.47	68.54(2)	53.43(1)	61.12(0.10)	64.77(8)	**71.32**$(\mathbf{1}, \mathbf{2})$
CONTROL CHART	600	60	6	81.17	81.17	82.94(1)	77.72(1)	79.45(0.90)	82.79(32)	**84.22**$(\mathbf{0.6}, \mathbf{32})$
PARKINSON	185	22	2	21.96	19.13	28.89(1)	27.36(13)	25.82(0.95)	28.89(2)	**36.08**$(\mathbf{0.5}, \mathbf{4})$
VERTEBRAL	310	6	3	39.26	39.26	52.06 (2)	41.76(2)	56.63(0.80)	49.13(8)	**59.57**$(\mathbf{0.1}, \mathbf{4})$
BREAST TISSUE	106	9	6	54.03	54.43	54.04(2)	49.33(2)	51.64(0.20)	54.18(32)	**55.29**$(\mathbf{0.6}, \mathbf{16})$
SEEDS	210	7	3	73.90	73.90	74.89(1)	73.06(1)	74.80(0.80)	70.79(32)	**75.52**$(\mathbf{0}, \mathbf{8})$
IMAGE SEG.	2310	19	7	67.06	67.41	67.42(1)	64.77(1)	31.83(0.99)	69.58 (2^{14})	**70.70**$(\mathbf{0.5}, \mathbf{64})$
YEAST	1484	8	10	30.58	31.11	31.37(2)	28.89(1)	27.50(0.90)	32.62(16)	**33.83**$(\mathbf{0.1}, \mathbf{16})$
AVERAGE	–	–	–	58.29	58.34	59.92	54.77	56.37	60.01	63.92

To further evaluate the P−RWDKC framework, instead of using the ground truth and cross-validation, here we choose the parameter values that optimize the Calinski-Harabasz (CH). We first compute the set of candidate partitions, one for each (α, t_d) combination in the considered parameter grid. Then we select as best the model with the highest CH and compute its NMI. We operate in the same way for the methods that have parameters (i.e. all but SC−SYM$_1$, SC−SYM$_2$, and PIC whose results are the same as in Tab. 3. The comparative results are shown in Tab. 4. Notice that P−RWDKC outperforms significantly the other methods in nearly all cases (see also the average performance in the last row of the table.). Compared to Tab. 3 here the NMI of P−RWDKC stays just a little lower. This indicates that the unsupervised tuning of the model parameters offers comparable graph partition quality to the previous case where we applied cross-validation using the ground truth.

Table 4: Clustering performance (NMI) on UCI datasets with estimated parameters (shown in brackets) according to the CH index.

DATASET	N	d	k	SC−SYM$_1$	SC−SYM$_2$	DI−SIM$_\mathrm{L}(\tau)$	DI−SIM$_\mathrm{R}(\tau)$	DSC+(γ)	PIC(t_d)	P−RWDK(α, t_d)
IRIS	150	4	3	80.58	80.58	74.98(1)	66.57(1)	68.63(0.80)	78.32(32)	**90.11**$(\mathbf{0.6}, \mathbf{4})$
GLASS	214	9	6	38.59	38.92	37.39 (2)	35.87 (1)	36.58(0.80)	42.79(128)	**42.13**$(\mathbf{0.3}, \mathbf{256})$
WINE	178	13	3	86.33	86.33	83.66(1)	85.62(1)	**91.09**(0.80)	86.33(4)	84.73$(0.9, 32)$
WBDC	569	30	2	67.73	69.47	68.54(2)	53.43(1)	61.12(0.10)	64.77(8)	**70.24**$(\mathbf{0.5}, \mathbf{2})$
CONTROL CHART	600	60	6	81.17	81.17	82.94(2)	77.44(1)	79.45(0.90)	82.79(32)	**83.19**$(\mathbf{0.2}, \mathbf{32})$
PARKINSON	185	22	2	21.96	19.13	28.89(1)	27.36(13)	22.97(0.95)	28.89(2)	**36.08**$(\mathbf{0.7}, \mathbf{4})$
VERTEBRAL	310	6	3	39.26	39.26	45.89 (2)	39.62(1)	54.24(0.80)	49.13(8)	**59.34**$(\mathbf{0.3}, \mathbf{4})$
BREAST TISSUE	106	9	6	54.03	54.43	54.04(2)	49.27(2)	51.64(0.20)	54.18(32)	**54.90**$(\mathbf{0.2}, \mathbf{16})$
SEEDS	210	7	3	73.90	73.90	**74.89**(**1**)	73.06(1)	74.80(0.80)	70.79(32)	**74.89**$(\mathbf{0.1}, \mathbf{8})$
IMAGE SEG.	2310	19	7	67.06	67.41	67.42(1)	64.77(1)	31.46(0.99)	69.58 (2^{14})	**68.79**$(\mathbf{0.1}, \mathbf{64})$
YEAST	1484	8	10	30.58	31.11	31.22(1)	28.89(1)	27.47(0.90)	32.62(16)	**33.60**$(\mathbf{0}, \mathbf{16})$
AVERAGE	–	–	–	58.29	58.34	58.82	54.57	51.95	60.01	63.45

Table 5: Clustering performance (NMI) on real-world datasets with optimal parameters in brackets.

DATASET	N	k	SC-SYM$_1$	RSC(τ)	PIC(t_d)	P-RWDKC(α, t_d)
POLITICAL BLOGS (ADAMIC & GLANCE, 2005)	1222	2	1.74	73.25(0.2)	57.37(8)	$\mathbf{74.02}(\mathbf{0.1}, \mathbf{32})$
CORA (BOJCHEVSKI & GÜNNEMANN, 2017)	2485	7	17.10	36.07(2.5)	7.86(4)	$\mathbf{49.71}(\mathbf{0.2}, \mathbf{16})$
COLLEGE FOOTBALL (GIRVAN & NEWMAN, 2002)	115	12	29.97	30.79(2.9)	29.98(2)	$\mathbf{30.79}(\mathbf{0.9}, \mathbf{2})$
KARATE CLUB (ZACHARY, 1977)	34	2	73.24	**83.72 (1.7)**	$\mathbf{83.72}(\mathbf{1})$	$\mathbf{83.72}(\mathbf{0}, \mathbf{1})$