Step-by-Step Community Detection for Volume-Regular Graphs

Luca Becchetti
Sapienza Universit di Roma
Rome, Italy
becchetti@diag.uniroma1.it

Emilio Cruciani
Gran Sasso Science Institute
L'Aquila, Italy
emilio.cruciani@gssi.it

Francesco Pasquale*
Universit di Roma Tor Vergata
Rome, Italy
pasquale@mat.uniroma2.it

Sara Rizzo
Gran Sasso Science Institute
L'Aquila, Italy
sara.rizzo@gssi.it

Abstract

Spectral techniques have proved amongst the most effective approaches to graph clustering. However, in general they require explicit computation of the main eigenvectors of a suitable matrix (usually the Laplacian matrix of the graph).

Recent work (e.g., Becchetti et al., SODA 2017) suggests that observing the temporal evolution of the power method applied to an initial random vector may, at least in some cases, provide enough information on the space spanned by the first two eigenvectors, so as to allow recovery of a hidden partition without explicit eigenvector computations. While the results of Becchetti et al. apply to perfectly balanced partitions and/or graphs that exhibit very strong forms of regularity, we extend their approach to graphs containing a hidden k partition and characterized by a milder form of volume-regularity. We show that the class of k-*volume regular* graphs is the largest class of undirected (possibly weighted) graphs whose transition matrix admits k "stepwise" eigenvectors (i.e., vectors that are constant over each set of the hidden partition). To obtain this result, we highlight a connection between volume regularity and lumpability of Markov chains. Moreover, we prove that if the stepwise eigenvectors are those associated to the first k eigenvalues and the gap between the k-th and the $(k+1)$-th eigenvalues is sufficiently large, the Averaging dynamics of Becchetti et al. recovers the underlying community structure of the graph in logarithmic time, with high probability.

*Partly supported by the University of "Tor Vergata" under research programme "Mission: Sustainability" project ISIDE (grant no. E81I18000110005)

1 Introduction

Clustering a graph in a way that reflects underlying community structure is a very important mining task [For10]. Informally speaking, in the classical setting, we are given a possibly weighted graph G and an integer k. Our goal is to partition the vertex set of $G = (V, E)$ into k disjoint subsets, so that the k induced subgraphs have high inner and low outer expansion. Spectral techniques have proved amongst the most effective approaches to graph clustering [NJW02, SM00, VL07]. The general approach to spectral graph clustering [VL07] normally implies embedding the vertices of G into the k-dimensional subspace spanned by the main k eigenvectors of a matrix defined in terms of G's adjacency matrix, typically its (normalized) Laplacian. Intuitively, one expects that, for a well-clustered graph with k communities, the profiles of the first k eigenvectors are correlated with the underlying community structure of G. Recent work has provided theoretical support to this approach. In particular, [LGT14] showed that, given the first k orthonormal eigenvectors of the normalized Laplacian, it is possible to produce a k-partition of the vertex set, corresponding to k suitably-defined indicator vectors, such that the associated values of the Rayleigh quotient are relatively small. More recently, [PSZ17] proved that, under suitable hypotheses on the spectral gap between the k-th and $(k+1)$-th eigenvalue of the normalized Laplacian of G, the span of the first k eigenvectors largely overlaps with the span of $\{D^{\frac{1}{2}}\boldsymbol{g}_1, \ldots, D^{\frac{1}{2}}\boldsymbol{g}_k\}$, where $D^{\frac{1}{2}}$ is the diagonal degree matrix of G, while the $\boldsymbol{g}_i$'s are indicator vectors describing a k-way partition $\{S_i\}_{i=1}^k$ of V, such that, for every i, the conductance of S_i is at most the k-way expansion constant $\rho(k)$ [LGT14]. Note that, if $\boldsymbol{v}$ is an eigenvector associated to the i-th smallest eigenvalue of the normalized Laplacian, $D^{-\frac{1}{2}}\boldsymbol{v}$ is an eigenvector corresponding to the i-th largest eigenvalue of the random walk transition matrix associated to G. Hence, when G is well-clustered, one might reasonably expect the first k eigenvectors of P reflect the community structure of G exhibiting almost-"stepwise" profiles, i.e., components relative to the same community are close. The aforementioned spectral approaches require explicit computation of the k main eigenvectors of a (generally symmetric) matrix.

In [BCN+17], the authors considered the case $k = 2$ for which they proposed the following distributed algorithm (Averaging dynamics, Algorithm 1): "At the outset, every node picks an initial value, independently and uniformly at random in $\{-1, 1\}$; then, in each synchronous round, every node updates its value to the average of those held by its neighbors. A node also tags itself `blue` if the last update increased its value, `red` otherwise" [BCN+17]. The authors showed that, under a variety of graph models exhibiting sparse balanced cuts, including the *stochastic block model* [HLL83], the process resulting from the above simple local rule converges, in logarithmic time, to a coloring that exactly or approximately (depending on the model) reflects the underlying cut. They further elaborated on how to extend the proposed approach to the case of multiple communities, providing an analysis for a strongly regular version of the stochastic block model with multiple communities. While results like [LGT14, PSZ17] provide further theoretical justification for spectral clustering, the approach proposed in [BCN+17] suggests that observing the temporal evolution of the power method applied to an initial random vector may, at least in some cases, provide equivalent information, without requiring explicit eigenvector computations.

1.1 Our contributions

The goal of this work is to take a further step in this direction by considering a more general class of graphs, even if still relatively "regular", with respect to that considered in [BCN+17]. The analysis of the Averaging dynamics on this class is considerably harder, but it is likely

to provide insights into the challenges of analyzing the general case, without all the intricacies of the latter. Our contribution is as follows:

- We define the class of *k-volume regular* graphs. This class of edge-weighted graphs includes those considered in [BCN+17] and it is the largest class of undirected, possibly weighted graphs that admit k "stepwise" eigenvectors (i.e., having constant values over the k steps that identify the hidden partition). This result uses a connection between volume regularity and lumpability of Markov chains [KS60, TK06].

- If the stepwise eigenvectors are those associated to the first k eigenvalues and the gap between the k-th and the $(k+1)$-th eigenvalues is sufficiently large, we show that running the AVERAGING dynamics for a suitable number of steps allows to recover the underlying community structure of the graph, with high probability.[1] To prove this, we provide a family of mutually orthonormal vectors which, when the graph is volume regular, span the eigenspace of the main k eigenvectors of the normalized adjacency matrix of the graph. It should be noted that the first and second of these vectors are respectively the main eigenvector and the Fiedler vector [Fie89] associated to the normalized adjacency matrix.

- While the results of [BCN+17] apply when the underlying communities are of the same size, our results do not require this assumption and they apply to weighted graphs. It should also be noted that volume regularity does not imply regularity of the graph in general, it is a weaker notion.

- We further show that variants of the AVERAGING dynamics (and/or its labeling rule) can address different problems (e.g., identifying bipartiteness) and/or other graph classes.

We finally note that the overall algorithm we consider can be viewed as a fully decentralized, synchronous algorithm that works in *anonymous* networks,[2] with a completely local clustering criterion, though it cannot be considered a *dynamics* in the sense of [BCN+17] since it requires a bound on the number of nodes in the underlying network.

1.2 Further related work

We briefly discuss further work that bears some relationship to this paper, either because adopting simple and/or decentralized heuristics to uncover community structure, or because relying on the use of spectral techniques.

Decentralized heuristics for block reconstruction. *Label propagation algorithms* [RAK07] are dynamics based on majority updating rules [AAE08] and have been applied for detecting communities in complex networks. Several papers present experimental results for such protocols on specific classes of clustered graphs [BC09, LM10, RAK07]. The only available rigorous analysis of a label propagation algorithm on planted partition graphs is the one presented in [KPS13], where the authors analyze a label propagation algorithm on $\mathcal{G}_{2n,p,q}$ graphs in the case of dense topologies. In particular, their analysis considers the case where $p = \Omega(1/n^{\frac{1}{4}-\epsilon})$ and $q = \mathcal{O}(p^2)$, a parameter range in which very dense clusters of constant diameter separated by a sparse cut occur w.h.p. In this setting, characterized by a polynomial gap between p and q, simple combinatorial and concentration arguments show that the protocol converges in constant expected time. They also conjecture a logarithmic bound for sparser topologies.

[1] An event $\mathcal{E}_n$ holds *with high probability* (*w.h.p.*) if $\mathbf{P}(\mathcal{E}_n) = 1 - \mathcal{O}(n^{-\gamma})$, for some constant $\gamma > 0$.

[2] I.e., nodes do not possess distinguished identities.

Following [BCN+17], a number of recent papers analyze simple distributed algorithms for community detection that rely on elementary dynamics. In the AVERAGING dynamics considered in this paper, every node communicates in parallel with all its neighbors in each round. While this might be too expensive in scenarios characterized by dense topologies, it is simply unfeasible in other settings (for instance, when links represent opportunistic meetings that occur asynchronously). Motivated by similar considerations, a first line of follow-up work considered "sparsified", asynchronous variants of the AVERAGING dynamics [BCM+18, MMM18, SZ17].

Another interesting direction is the rigorous analysis of well-known (non-linear) dynamics based on *majority rules* on graphs that exhibit community structure. In [CNNS18], Cruciani et al. consider the *2-Choices* dynamics where, in each round, every node picks two random neighbors and updates its value to the most frequent among its value and those held by its sampled neighbors. They show that if the underlying graph has a suitable core-periphery structure and the process starts in a configuration where nodes in core and periphery have different states, the system either rapidly converges to the core's state or reaches a metastable regime that reflects the underlying graph structure. Similar results have been also obtained for clustered regular graphs with dense communities in [CNS19], where the *2-Choices* dynamics is proposed as a distributed algorithm for community detection.

Although based on the AVERAGING dynamics and thus extremely simple and fully decentralized, the algorithm we consider in this paper is not itself a dynamics in the sense proposed in [BCN+17], since its clustering criterion is applied within a time window, which in turn requires (at least approximate) knowledge of the network size.

Because of their relevance for the reconstruction problem, we also briefly discuss the class of *belief propagation algorithms*, best known as message-passing algorithms for performing inference in graphical models [Mac03]. Though not a dynamics, Belief propagation is still a simple approach. Moreover, there is non-rigorous, strong supporting evidence that some *belief propagation algorithms* might be optimal for the reconstruction problem [DKMZ11]. A rigorous analysis is a major challenge; in particular, convergence to the correct value of belief propagation is far from being fully-understood on graphs which are not trees [MK07, Wei00]. As we discuss in the next subsection, more complex algorithms inspired by belief propagation, have been rigorously shown to perform reconstruction optimally.

General algorithms for block reconstruction. Several algorithms for community detection are *spectral*: They typically consider the eigenvector associated to the second largest eigenvalue of the adjacency matrix A of G, or the eigenvector corresponding to the largest eigenvalue of the matrix $A - \frac{d}{n}J$ [Bop87, CO05, CO10, McS01],[3] since these are correlated with the hidden partition. More recently spectral algorithms have been proposed [AS15, BLM15, CO10, KMM+13, MNS13, PSZ17] that find a weak reconstruction even in the sparse, tight regime.

Interestingly, spectral algorithms turn out to be a feasible approach also in distributed settings. In particular, Kempe and McSherry [KM04] show that eigenvalue computations can be performed in a distributed fashion, yielding distributed algorithms for community detection under various models, including the stochastic block model. However, their algorithm does not match any simple decentralized computing model. In particular, the algorithm of Kempe and McSherry as well as any distributed version of the above mentioned centralized algorithms are neither dynamics, nor do they correspond to the notion of *light-weight* algorithm of Hassin and Peleg [HP01]. Moreover, the mixing time of the simple random walk on the graph is a bottleneck for the distributed algorithm of Kempe and McSherry and for any algorithm that performs community detection in a graph G by employing the power method or the Lanczos

[3] A is the adjacency matrix of G, J is the matrix having all entries equal to 1, d is the average degree, and n is the number of vertices.

method [Lan50] as a subroutine. This is not the case for the AVERAGING dynamics, since it removes the component of the state in the span of the main eigenvector.

In general, the reconstruction problem has been studied extensively using a multiplicity of techniques, which include combinatorial algorithms [DF89], belief propagation [DKMZ11] and variants of it [MNS16], spectral-based techniques [CO10, McS01], Metropolis approaches [JS98], and semidefinite programming [ABH14], among others.

2 Preliminaries

Notation. Consider an undirected edge-weighted graph $G = (V, E, w)$ with nonnegative weights. For each node $u \in V$, we denote by $\delta(u)$ the *volume*, or *weighted degree*, of node u, namely $\delta(u) = \sum_{v:(u,v)\in E} w(u, v)$. D denotes the diagonal matrix, such that $D_{uu} = \delta(u)$ for each $u \in V$. Without loss of generality we assume $\min_u \delta(u) = 1$, since the behavior of the AVERAGING dynamics (and the corresponding analysis) is not affected by a normalization of the weights. We refer to the maximum volume of a node as $\Delta := \max_u \delta(u)$.

In the remainder, W denotes the *weighted adjacency matrix* of G, while $P = D^{-1}W$ is the *transition matrix* of a random walk on G, in which a transition from node u to node v occurs with probability proportional to $w(u, v)$. We call $\lambda_1, \ldots, \lambda_n$ the eigenvalues of P, in non-increasing order, and $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n$ a family of eigenvectors of P, such that $P\boldsymbol{v}_i = \lambda_i \boldsymbol{v}_i$. We let $N = D^{-\frac{1}{2}} W D^{-\frac{1}{2}} = D^{\frac{1}{2}} P D^{-\frac{1}{2}}$ denote the *normalized weighted adjacency matrix* of G. Note that N is symmetric and that its spectrum is the same as that of P. We denote by $\boldsymbol{w}_1, \ldots, \boldsymbol{w}_n$ a family of eigenvectors of N, such that $P\boldsymbol{w}_i = \lambda_i \boldsymbol{w}_i$. It is important to note that $\boldsymbol{w_i}$ is an eigenvector of N if and only if $D^{-\frac{1}{2}}\boldsymbol{w}_i$ is an eigenvector of P.

2.1 Averaging dynamics

The simple algorithm we consider in this paper, named AVERAGING dynamics (Algorithm 1) after [BCM+18] in which the algorithm was first proposed, can be seen as an application of the power method, augmented with a Rademacher initialization and a suitable labeling scheme. In this form, it is best described as a distributed process, executed by the nodes of an underlying edge-weighted graph. The AVERAGING dynamics can be used as a building-block to achieve "community detection" in some classes of "regular" and "almost regular" graphs. Herein, we extend its use and analysis to broader graph classes and, in one case, to a different problem.

Algorithm 1 AVERAGING dynamics

Rademacher initialization: At round $t = 0$, every node $v \in V$ independently samples its value $\boldsymbol{x}^{(0)}(v)$ from $\{-1, +1\}$ uniformly at random.

Update rule: At each subsequent round $t \geqslant 1$, every node $v \in V$:

1. *Averaging*: updates its value $\boldsymbol{x}^{(t)}(v)$ to the weighted average of the values of its neighbors at the end of the previous round.
2. *Labeling*: if $\boldsymbol{x}^{(t)}(v) \geqslant \boldsymbol{x}^{(t-1)}(v)$ then v sets $\texttt{label}^{(t)}(v) = 1$; otherwise v sets $\texttt{label}^{(t)}(v) = 0$.

Spectral decomposition of the transition matrix. Let $\boldsymbol{x}^{(t)}$ denote the *state vector* at time t, i.e., the vector whose u-th entry is the value held by node u at time t. We let $\boldsymbol{x}^{(0)} = \boldsymbol{x}$ denote the initial state vector. Globally, the *averaging* update rule of Algorithm 1 corresponds

to one iteration of the power method, in this case an application of the transition matrix P to the current state vector, i.e., $\boldsymbol{x}^{(t)} = P\boldsymbol{x}^{(t-1)}$. We can write

$$\boldsymbol{x}^{(t)} = P^t\boldsymbol{x} = D^{-\frac{1}{2}}N^tD^{\frac{1}{2}}\boldsymbol{x} \overset{(a)}{=} D^{-\frac{1}{2}}\sum_{i=1}^{n}\lambda_i^t\boldsymbol{w}_i\boldsymbol{w}_i^\intercal\sum_{i=1}^{n}\beta_i\boldsymbol{w}_i = \sum_{i=1}^{n}\lambda_i^t\beta_iD^{-\frac{1}{2}}\boldsymbol{w}_i,$$

where in (a) we spectrally decomposed the matrix N^t and expressed the vector $D^{\frac{1}{2}}\boldsymbol{x}$ as a linear combination of the eigenvectors of N, i.e., $D^{\frac{1}{2}}\boldsymbol{x} = \sum_{i=1}^{n}\beta_i\boldsymbol{w}_i$, with $\beta_i = \langle D^{\frac{1}{2}}\boldsymbol{x}, \boldsymbol{w}_i\rangle$. By explicitly writing the β_is and by noting that $\boldsymbol{w}_i = \frac{D^{\frac{1}{2}}\boldsymbol{v}_i}{\|D^{\frac{1}{2}}\boldsymbol{v}_i\|}$ we conclude that

$$\boldsymbol{x}^{(t)} = \sum_{i=1}^{n}\lambda_i^t\frac{\langle D^{\frac{1}{2}}\boldsymbol{x}, D^{\frac{1}{2}}\boldsymbol{v}_i\rangle}{\|D^{\frac{1}{2}}\boldsymbol{v}_i\|}D^{-\frac{1}{2}}\frac{D^{\frac{1}{2}}\boldsymbol{v}_i}{\|D^{\frac{1}{2}}\boldsymbol{v}_i\|} = \sum_{i=1}^{n}\lambda_i^t\alpha_i\boldsymbol{v}_i, \tag{1}$$

where $\alpha_i := \frac{\langle D^{\frac{1}{2}}\boldsymbol{x}, D^{\frac{1}{2}}\boldsymbol{v}_i\rangle}{\|D^{\frac{1}{2}}\boldsymbol{v}_i\|^2} = \frac{\boldsymbol{x}^\intercal D\boldsymbol{v}_i}{\|D^{\frac{1}{2}}\boldsymbol{v}_i\|^2}$.

Note that $\lambda_1 = 1$ and $\boldsymbol{v}_1 = \mathbf{1}$,[4] since P is stochastic and, if G is connected and non bipartite, $\lambda_i \in (-1, 1)$ for every $i > 1$. The long term behavior of the dynamics can be written as

$$\lim_{t\to\infty}\boldsymbol{x}^{(t)} = \lim_{t\to\infty}\sum_{i=1}^{n}\lambda_i^t\alpha_i\boldsymbol{v}_i = \alpha_1\mathbf{1}, \text{ with } \alpha_1 = \frac{\sum_{u\in V}\delta(u)\boldsymbol{x}(u)}{\sum_{u\in V}\delta(u)} = \sum_{u\in V}\frac{\delta(u)}{\mathrm{vol}(V)}\boldsymbol{x}(u),$$

i.e., each node converges to the initial global weighted average of the network.

2.2 Community-sensitive algorithms

We give the following definition of *community sensitive algorithm*, that closely resembles that of locality-sensitive hashing (see, e.g., [LRU14]).

Definition 2.1 (Community-sensitive algorithm)**.** *Let $\mathcal{A}$ be a randomized algorithm that takes in input a (possibly weighted) graph $G = (V, E)$ with a hidden partition $\mathcal{V} = \{V_1, \ldots, V_k\}$ and assigns a Boolean value $\mathcal{A}(G)[v] \in \{0, 1\}$ to each node $v \in V$. We say $\mathcal{A}$ is an (ε, δ)-Community-sensitive algorithm, for some $\varepsilon, \delta > 0$, if the following two conditions hold:*

1. *For each set V_i of the partition and for each pair of nodes $u, v \in V_i$ in that set, the probability that the algorithm assigns the same Boolean value to u and v is at least $1 - \varepsilon$,*

$$\forall i \in [k],\ \forall u, v \in V_i,\ \mathbf{P}\left(\mathcal{A}(G)[u] = \mathcal{A}(G)[v]\right) \geqslant 1 - \varepsilon.$$

2. *For each pair V_i, V_j of distinct sets of the partition and for each pair of nodes $u \in V_i$ and $v \in V_j$, the probability that the algorithm assigns the same value to u and v is at most δ,*

$$\forall i, j \in [k] \text{ with } i \neq j,\ \forall u \in V_i, \forall v \in V_j,\ \mathbf{P}\left(\mathcal{A}(G)[u] = \mathcal{A}(G)[v]\right) \leqslant \delta.$$

For example, for $(\varepsilon, \delta) = (1/n, 1/2)$, an algorithm that simply assigns the same value to all nodes would satisfy the first condition but not the second one, while an algorithm assigning 0 or 1 to each node with probability 1/2, independently of the other nodes, would satisfy the second condition but not the first one.

Note that Algorithm 1 is a distributed algorithm that, at each round t, assigns one out of two labels to each node of a graph. In the next section (see Theorem 4.1) we prove that a time window $[T_1, T_2]$ exists, such that for all rounds $t \in [T_1, T_2]$, the assignment of the AVERAGING dynamics satisfies both conditions in Definition 2.1: The first condition with $\varepsilon = \varepsilon(n) = \mathcal{O}(n^{-\frac{1}{2}})$, the second with $\delta = \mathcal{O}(1)$.

[4] Here and in the remainder, $\mathbf{1}$ denotes the vector whose entries are 1.

Community-sensitive labeling. If we execute $\ell = \Theta(\log n)$ independent runs of an (ε, δ)-Community-sensitive algorithm $\mathcal{A}$, each node is assigned a *signature* of ℓ binary values, with pairwise Hamming distances probabilistically reflecting community membership of the nodes. More precisely, let $\mathcal{A}$ be an (ε, δ)-Community-sensitive algorithm and let $\mathcal{A}_1, \ldots, \mathcal{A}_\ell$ be $\ell = \Theta(\log n)$ independent runs of $\mathcal{A}$. For each node $u \in V$, let $\boldsymbol{s}(u) = (s_1(u), \ldots, s_\ell(u))$ denote the *signature* of node u, where $s_i(u) = \mathcal{A}_i(G)[u]$. For each pair nodes u, v, let $h(u,v) = |\{i \in [\ell] : s_i(u) \neq s_i(v)\}|$ be the Hamming distance between $\boldsymbol{s}(u)$ and $\boldsymbol{s}(v)$. The following lemma follows from a straightforward application of Chernoff bounds.

Lemma 2.1 (From Community-sensitive algorithm to Community-sensitive labeling)**.** *Let $\mathcal{A}$ be an (ε, δ)-Community-sensitive algorithm with $\varepsilon = o(1)$ and $\delta = \mathcal{O}(1)$. For large enough $\ell = \Theta(\log n)$, two positive constants α, β exist, with $0 \leqslant \alpha < \beta \leqslant 1$, such that for each pair of nodes $u, v \in V$ it holds that:*

1. *If u and v belong to the same community then $h(u,v) \leqslant \alpha\ell$, w.h.p.*
2. *If u and v belong to different communities then $h(u,v) \geqslant \beta\ell$, w.h.p.*

Proof. If u and v belong to the same community, then $\mathbf{E}[h(u,v)] \leqslant \varepsilon\ell$. If they belong to different communities, then $\mathbf{E}[h(u,v)] \geqslant (1-\delta)\ell$. The thesis follows by a standard application of Chernoff bounds, e.g., by choosing $\alpha = (1-\delta)/4$ and $\beta = (1-\delta)/2$. □

2.3 Volume-regular graphs

Recall that, for an undirected edge-weighted graph $G = (V, E, w)$, we denote by $\delta(u)$ the volume a node $u \in V$, i.e., $\delta(u) = \sum_{v:(u,v)\in E} w(u,v)$. Note that the transition matrix P of a random walk on G is such that $P_{uv} = w(u,v)/\delta(u)$. Given a partition $\mathcal{V} = \{V_1, \ldots, V_k\}$ of the set of nodes V, for a node $u \in V$ and a partition index $i \in [k]$, $\delta_i(u)$ denotes the overall weight of edges connecting u to nodes in V_i, $\delta_i(u) = \sum_{v\in V_i\,:\,u,v\in E} w(u,v)$. Hence, $\delta(u) = \sum_{i=1}^{k} \delta_i(u)$.

Definition 2.2 (Volume-regular graph)**.** *Let $G = (V, E, w)$ be an undirected edge-weighted graph with $|V| = n$ nodes and let $\mathcal{V} = \{V_1, \ldots, V_k\}$ be a k-partition of the nodes, for some $k \in [n]$. We say that G is* volume regular *with respect to $\mathcal{V}$ if, for every pair of partition indexes $i, j \in [k]$ and for every pair of nodes $u, v \in V_i$, $\frac{\delta_j(u)}{\delta(u)} = \frac{\delta_j(v)}{\delta(v)}$. We say that G is k-volume regular if there exists a k-partition $\mathcal{V}$ of the nodes such that G is volume regular with respect to $\mathcal{V}$.*

In other words, G is volume regular if there exists a partition of the nodes such that the fraction of a node's volume toward a set of the partition is constant across nodes of the same set. Note that all graphs with n nodes are trivially 1- and n-volume regular.

Let $G = (V, E, w)$ be a k-volume regular graph and let P be the transition matrix of a random walk on G. In the next lemma we prove that the span of k linearly independent eigenvectors of P equals the span of the indicator vectors of the k communities of G. The proof makes use of the correspondence between random walks on volume regular graphs and *ordinary lumpable* Markov chains [KS60]; in particular the result follows from Lemma 3.1 and Lemma 3.2, that we prove in Section 3.

Lemma 2.2. *Let P be the transition matrix of a random walk on a k-volume regular graph $G = (V, E, w)$ with k-partition $\mathcal{V} = \{V_1, \ldots, V_k\}$. There exists a family $\{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k\}$ of linearly independent eigenvectors of P such that $Span(\{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k\}) = Span(\{\mathbf{1}_{V_1}, \ldots, \mathbf{1}_{V_k}\})$, with $\mathbf{1}_{V_i}$ the indicator vector of the i-th set of the partition, for $i \in [k]$.*

In the rest of the paper we call "stepwise" the eigenvectors of P that can be written as linear combinations of the indicator vectors of the communities. In the next definition, we formalize the fact that a k-volume regular graph is *clustered* if the k linearly independent stepwise eigenvectors of P, whose existence is guaranteed by the above lemma, are associated to the k largest eigenvalues of P.

Definition 2.3 (Clustered volume regular graph)**.** *Let $G = (V, E, w)$ be a k-volume regular graph and let P be the transition matrix of a random walk on G. We say that G is a* clustered k-volume regular graph *if the k stepwise eigenvectors of P are associated to the first k largest eigenvalues of P.*

3 Volume-regular graphs and lumpable Markov chains

The class of volume-regular graphs is deeply connected with the definition of *lumpability* [KS60] of Markov chains. We here first recall the definition of lumpable Markov chain and then show that a graph G is volume-regular if and only if the associated weighted random walk is a lumpable Markov chain.

Definition 3.1 (Ordinary lumpability of Markov Chains)**.** *Let $\{X_t\}_t$ be a finite Markov chain with state space V and transition matrix $P = (P_{uv})_{u,v \in V}$ and let $\mathcal{V} = \{V_1, \ldots, V_k\}$ be a partition of the state space. Markov chain $\{X_t\}_t$ is* ordinary lumpable *with respect to $\mathcal{V}$ if, for every pair of partition indexes $i, j \in [k]$ and for every pair of nodes in the same set of the partition $u, v \in V_i$, it holds that*

$$\sum_{w \in V_i} P_{uw} = \sum_{w \in V_i} P_{vw}, \quad \forall\ u, v \in V_j. \tag{2}$$

We define the lumped matrix $\widehat{P}$ *of the Markov Chain as the matrix such that $\widehat{P}_{ij} = \sum_{w \in V_i} P_{uw}$, for any $u \in V_j$.*

We first prove that random walks on Volume-regular graphs define exactly the subset of reversible and ordinary lumpable Markov chains.

Lemma 3.1. *A reversible Markov chain $\{X_t\}_t$ is ordinary lumpable if and only if it is a random walk on a volume-regular graph.*

Proof. Assume first that $\{X_t\}_t$ is ordinary lumpable and let P be the corresponding transition matrix. Consider the weighted graph $G = (V, E, w)$ obtained from P as follows: V corresponds to the set of states in P, while $w(u, v) = \pi(u)P_{uv}$, for every $u, v \in V$, with π the stationary distribution of P. Note that G is an undirected graph, i.e., $w(u, v) = \pi(u)P_{uv} \overset{(a)}{=} \pi(v)P_{vu} = w(v, u)$, where (a) holds because P is reversible. Moreover

$$\delta(u) = \sum_{z \in V} w(u, v) = \sum_{z \in V} \pi(u)P_{uv} = \pi(u) \sum_{z \in V} P_{uv} \overset{(a)}{=} \pi(u),$$

where (a) holds because P is stochastic. Thus G meets Definition 2.2 because, for any $u, v \in V_i$,

$$\frac{\delta_j(u)}{\delta(u)} = \frac{1}{\pi(u)} \sum_{z \in V_j} w(u, z) = \sum_{z \in V_j} P_{uz} = \sum_{z \in V_j} P_{vz} = \frac{1}{\pi(v)} \sum_{z \in V_j} w(v, z) = \frac{\delta_j(v)}{\delta(v)}.$$

Next, assume G is k-volume-regular with respect to the partition $\mathcal{V} = \{V_1, \ldots, V_k\}$. Let P be the transition matrix of the corresponding random walk. For every $i, j \in [k]$ and for every

$u, v \in V_i$ we have:

$$\sum_{z \in V_j} P_{uz} = \sum_{z \in V_j} \frac{w(u,z)}{\delta(u)} = \frac{\delta_j(u)}{\delta(u)} \overset{(a)}{=} \frac{\delta_j(v)}{\delta(v)} = \sum_{z \in V_j} \frac{w(v,z)}{\delta(v)} = \sum_{z \in V_j} P_{vz},$$

where (a) follows from Definition 2.2. Moreover note that P is reversible with respect to distribution π, where $\pi(u) = \frac{\delta(u)}{\mathrm{vol}(G)}$. □

Note that infinitely many k-volume-regular graphs have the same k-ordinary lumpable random walk chain.

We next show that a Markov chain is k-ordinary lumpable if and only if the corresponding transition matrix P has k stepwise, linearly independent eigenvectors.

Lemma 3.2. *Let P be the transition matrix of a Markov chain. Then P has k stepwise linearly independent eigenvectors if and only if P is ordinary lumpable.*

Proof. We divide the proof in two parts. First, we assume that P is ordinary lumpable and show that P has k stepwise linearly independent eigenvectors. Second, we assume that P has k stepwise linearly independent eigenvectors and show that P is ordinary lumpable.

1. Let P be ordinary lumpable and $\widehat{P}$ its lumped matrix. Let $\lambda_i, \boldsymbol{v}_i$ be the eigenvalues and eigenvectors of $\widehat{P}$, for each $i \in [k]$. Let $\boldsymbol{w}_i \in \mathbb{R}^n$ be a stepwise vector defined as

$$\boldsymbol{w}_i = (\boldsymbol{v}_i(1), \ldots, \boldsymbol{v}_i(1),\ \boldsymbol{v}_i(2), \ldots, \boldsymbol{v}_i(2),\ \ldots,\ \boldsymbol{v}_i(k), \ldots, \boldsymbol{v}_i(k))^\intercal,$$

where $\boldsymbol{v}_i(j)$ indicates the j-th component of $\boldsymbol{v}_i$, and then the n_j components relative to V_j are all equal to $\boldsymbol{v}_i(j)$.

Since the eigenvectors $\boldsymbol{v}_i$ of $\widehat{P}$ are linearly independent, the vectors $\boldsymbol{w}_i$ are also linearly independent. Moreover, it is easy to see that $P\boldsymbol{w}_i = \lambda_i \boldsymbol{w}_i$ by just verifying the equation for every $i \in [k]$.

2. Assume P has k stepwise linearly independent eigenvectors $\boldsymbol{w}_i$, associated to k eigenvalues λ_i, for each $i \in [k]$. Let $\boldsymbol{v}_i \in \mathbb{R}^k$ the vector that has as components the k constant values in the steps of $\boldsymbol{w}_i$. Since the $\boldsymbol{w}_i$ are linearly independent, the $\boldsymbol{v}_i$ also are.

For every eigenvector $\boldsymbol{w}_i$ and for every two states $x, y \in V_l$, for every $l \in [k]$, we have that $\lambda_i \boldsymbol{w}_i(x) = \lambda_i \boldsymbol{w}_i(y)$ since $\boldsymbol{w}_i$ is stepwise. Then, since $P\boldsymbol{w}_i = \lambda_i \boldsymbol{w}_i$, we have that

$$\sum_{j=1}^{k} \sum_{z \in V_j} P_{xz} \boldsymbol{v}_i(j) = (P\boldsymbol{w}_i)(x) = (P\boldsymbol{w}_i)(y) = \sum_{j=1}^{k} \sum_{z \in V_j} P_{yz} \boldsymbol{v}_i(j).$$

Thus $\sum_{j=1}^{k} \boldsymbol{v}_i(j) \sum_{z \in V_j} (P_{xz} - P_{yz}) = 0$ and then it follows that

$$\sum_{j=1}^{k} \boldsymbol{v}_i(j)\, \boldsymbol{u}_{xy}(j) = \langle \boldsymbol{u}_{xy}, \boldsymbol{v}_i \rangle = 0,$$

where $\boldsymbol{u}_{xy}(j) = \sum_{z \in V_j} (P_{xz} - P_{yz})$. Since the $\boldsymbol{v}_i$ are k linearly independent vectors in a k-dimensional space, $\boldsymbol{u}_{xy}$ cannot be orthogonal to all of them and then it has to be the null vector, i.e. $\boldsymbol{u}_{xy}(j) = 0$ for all $j \in [k]$. This implies that P is ordinary lumpable, i.e. $\sum_{z \in V_j} P_{xz} = \sum_{z \in V_j} P_{yz}$. It is easy to verify that the eigenvalues and eigenvectors of $\widehat{P}$ are exactly $\lambda_i, \boldsymbol{v}_i$, with $i \in [k]$. □

4 Averaging dynamics on clustered volume regular graphs

For a volume-regular graph $G = (V, E, w)$ with n nodes and k-partition $\mathcal{V} = \{V_1, \ldots, V_k\}$ we name $N = \frac{\max_i |V_i|}{\min_i |V_i|}$ the ratio between the maximum and minimum sizes of the communities. In this section we prove the following result for volume regular graphs.

Theorem 4.1. *Let $G = (V, E, w)$ be a connected clustered k-volume-regular graph with n nodes and k-partition $\mathcal{V} = \{V_1, \ldots, V_k\}$, with $k \leqslant \sqrt{n}$, maximum weighted degree $\Delta \leqslant poly(n)$, and $N = \mathcal{O}(\sqrt{k}/\Delta)$. If $\lambda_k > \frac{1}{2}$ and $(1 - \lambda_2) \geqslant (\lambda_2 - \lambda_k)\Delta^{\frac{3}{2}} n^{1+c}$, for an arbitrarily-small positive constant c, then a time interval $[T_1, T_2]$ exists, with $T_1 = \mathcal{O}(\log n \,/\, \log(\lambda_k/\lambda_{k+1}))$ and $T_2 = \Omega(n^{c/3})$, such that for each time $t \in [T_1, T_2]$ the* Averaging *dynamics truncated at round t is a $(\mathcal{O}(n^{-\frac{1}{2}}), \mathcal{O}(1))$-community sensitive algorithm, w.h.p.*

In the remainder of this section, we first introduce further notation and then state the two main technical lemmas (Lemma 4.1 and Lemma 4.2), that will be used in the proof of Theorem 4.1, which concludes this section. Due to lack of space, here we only give a sketch of the proof of Lemma 4.1, while its full proof is deferred to Appendix B.

Let $G = (V, E, w)$ be a clustered k-volume regular graph and, without loss of generality, let $V_1, \ldots, V_k$ be an arbitrary ordering of its communities. We introduce a family of stepwise vectors that generalize Fiedler vector [Fie89], namely

$$\left\{ \boldsymbol{\chi}_i = \sqrt{\frac{\hat{m}_i}{m_i}} \mathbf{1}_{V_i} - \sqrt{\frac{m_i}{\hat{m}_i}} \mathbf{1}_{\hat{V}_i} \;:\; i \in [k-1] \right\},$$

where $\mathbf{1}_{V_i}$ is the indicator vector of the set V_i and, for convenience sake, we denoted by m_i the volume of the i-th community, $\hat{V}_i$ the set of all nodes in communities $i+1, \ldots, k$, and $\hat{m}_i$ the volume of $\hat{V}_i$, i.e., $m_i = \sum_{u \in V_i} \delta(u)$, $\hat{V}_i = \bigcup_{h=i+1}^{k} V_h$, and $\hat{m}_i = \sum_{h=i+1}^{k} m_h$. Note that vectors $\boldsymbol{\chi}_i$s are "stepwise" with respect to the communities of G (i.e., for every $i \in [k-1]$, $\boldsymbol{\chi}_i(u) = \boldsymbol{\chi}_i(v)$ whenever u and v belong to the same community).

Recall from Equation (1) that the initial state vector can be written as $\boldsymbol{x} = \sum_{i=1}^{n} \alpha_i \boldsymbol{v}_i$. Let $\boldsymbol{z} = \sum_{i=1}^{k} \alpha_i \boldsymbol{v}_i$ and note that $\boldsymbol{z} = \alpha_1 \mathbf{1} + \sum_{i=1}^{k-1} \gamma_i \boldsymbol{\chi}_i$ by applying Lemma 2.2 and because $Span\left(\{\mathbf{1}, \boldsymbol{\chi}_1, \ldots, \boldsymbol{\chi}_{k-1}\}\right) = Span\left(\{\mathbf{1}_{V_1}, \ldots, \mathbf{1}_{V_k}\}\right)$. Let us now define the vector $\boldsymbol{y} = \boldsymbol{z} - \alpha_1 \mathbf{1}$ or, equivalently,

$$\boldsymbol{y} = \sum_{i=1}^{k-1} \gamma_i \boldsymbol{\chi}_i, \text{ where } \gamma_i = \frac{\boldsymbol{x}^\intercal D \boldsymbol{\chi}_i}{\left\| D^{1/2} \boldsymbol{\chi}_i \right\|^2}.$$

Note that the coefficients γ_is are proportional to the length of the projection of the (inhomogeneously) contracted state vector on the (inhomogeneously) contracted, not anymore stepwise, $D^{\frac{1}{2}}\boldsymbol{\chi}_i$s and can be computed since the vectors in the family $\{D^{\frac{1}{2}}\mathbf{1}\} \cup \{D^{\frac{1}{2}}\boldsymbol{\chi}_i : i \in [k-1]\}$ are mutually orthogonal.[5]

The binary coloring of each node only depends on the difference of its state in two consecutive rounds (see Algorithm 1). Essentially in Lemma 4.1 we show that, under suitable assumptions on the transition matrix of a random walk on G, there exists a time window where the the difference of the state vector in two consecutive rounds, i.e., $\boldsymbol{x}^{(t)} - \boldsymbol{x}^{(t+1)}$, can be approximated by the previously defined vector $\boldsymbol{y}$ in a way that the sign of the two vectors is equal in any component, with high probability.

[5]The mutual orthogonality of the vectors, including $D^{\frac{1}{2}}\mathbf{1}$, is also one of the reasons why other "simpler" families of stepwise vectors, e.g., the indicator vectors of the communities, are not used instead.

Lemma 4.1 (Sign of the difference)**.** *Let $G = (V, E, w)$ be a clustered k-volume regular graph. If $\lambda_k > \frac{1}{2}$ and $(1 - \lambda_2) \geqslant (\lambda_2 - \lambda_k)\Delta^{\frac{3}{2}} n^{1+c}$, for an arbitrarily-small positive constant c, then a time interval $[T_1, T_2]$ exists, with $T_1 = \mathcal{O}(\log n \,/\, \log(\lambda_k/\lambda_{k+1}))$ and $T_2 = \Omega(n^{c/3})$, such that for each node $u \in V$ it holds that*

$$\operatorname{sgn}(\, \boldsymbol{x}^{(t)}(u) - \boldsymbol{x}^{(t+1)}(u)\,) = \operatorname{sgn}(\boldsymbol{y}(u))$$

for every round $t \in [T_1, T_2]$ of the execution of the Averaging *dynamics, w.h.p.*

Sketch of proof. Recall from Equation (1) that the state vector at time t, i.e., $\boldsymbol{x}^{(t)}$, can be written as the sum of the first k stepwise vectors of P and of the remaining ones, namely

$$\boldsymbol{x}^{(t)} = \alpha_1 \mathbf{1} + \sum_{i=2}^{k} \lambda_i^t \alpha_i \boldsymbol{v}_i + \sum_{i=k+1}^{n} \lambda_i^t \alpha_i \boldsymbol{v}_i = \alpha_1 \mathbf{1} + \boldsymbol{c}^{(t)} + \boldsymbol{e}^{(t)},$$

where we call $\boldsymbol{c}^{(t)} := \sum_{i=2}^{k} \lambda_i^t \alpha_i \boldsymbol{v}_i$ the *core contribution* and $\boldsymbol{e}^{(t)} := \sum_{i=k+1}^{n} \lambda_i^t \alpha_i \boldsymbol{v}_i$ the *error contribution*. If we look at the difference of the state vector between two consecutive rounds, for each node $u \in V$, the first term cancels out being constant over time and we get $\boldsymbol{x}^{(t)}(u) - \boldsymbol{x}^{(t+1)}(u) = \boldsymbol{c}^{(t)}(u) - \boldsymbol{c}^{(t+1)}(u) + \boldsymbol{e}^{(t)}(u) - \boldsymbol{e}^{(t+1)}(u)$. Note that the sign of the difference between two consecutive states of each node $u \in V$ is determined by the difference of the core contributions during the two consecutive rounds, i.e., $\boldsymbol{c}^{(t)}(u) - \boldsymbol{c}^{(t+1)}(u)$, whenever

$$\left|\boldsymbol{c}^{(t)}(u) - \boldsymbol{c}^{(t+1)}(u)\right| > \left|\boldsymbol{e}^{(t)}(u) - \boldsymbol{e}^{(t+1)}(u)\right|. \tag{3}$$

To find the conditions on t that make Eq. (3) hold, we give a bound to both the left and right hand side of the inequality. In detail:

1. We know from Lemma B.2 that $\left|\boldsymbol{c}^{(t)}(u) - \boldsymbol{c}^{(t+1)}(u)\right| > \frac{1}{2}\lambda_k^t (1 - \lambda_2) \left|\boldsymbol{y}(u)\right|$ for every $u \in V$ and for every time $t < T_2$, where $T_2 = \Omega(n^{\frac{c}{3}})$, since by hypothesis $\lambda_k > \frac{1}{2}$ and $(1 - \lambda_2) \geqslant (\lambda_2 - \lambda_k)\Delta^{\frac{3}{2}} n^{1+c}$.
2. We know from Lemma B.3 that $|\boldsymbol{e}^{(t)}(u)| \leqslant \lambda_{k+1}^t \sqrt{\Delta n}$, for every $u \in V$, and thus it follows that $\left|\boldsymbol{e}^{(t)}(u) - \boldsymbol{e}^{(t+1)}(u)\right| \leqslant \left|\boldsymbol{e}^{(t)}(u)\right| + \left|\boldsymbol{e}^{(t+1)}(u)\right| \leqslant 2\lambda_{k+1}^t \sqrt{\Delta n}$.

Combining Lemma B.2 and Lemma B.3, we get that if the following inequality holds, i.e.,

$$\frac{1}{2}\lambda_k^t (1 - \lambda_2) \left|\boldsymbol{y}(u)\right| > 2\lambda_{k+1}^t \sqrt{\Delta n}, \tag{4}$$

then also Equation (3) holds. By moving the terms dependent from t on the left hand side and by taking the logarithm of both sides, we can finally find the conditions on t such that Equation (4) is satisfied, i.e., all times $t > T_1$ where

$$T_1 = \log\left(\frac{4\sqrt{\Delta n}}{(1 - \lambda_2)\left|\boldsymbol{y}(u)\right|}\right) \cdot \frac{1}{\log\left(\frac{\lambda_k}{\lambda_{k+1}}\right)}.$$

Note that $T_1 = \mathcal{O}(\log n \,/\, \log(\frac{\lambda_k}{\lambda_{k+1}}))$ and that $T_1 = \mathcal{O}(\log n)$ when $\frac{\lambda_k}{\lambda_{k+1}} = \Omega(1)$. In fact:

1. We know by hypothesis that the maximum weighted degree of a node is at most polynomial in n, i.e., $\Delta \leqslant poly(n)$.

2. We know from the Cheeger's inequality for weighted graphs (Theorem A.2) the relation between the spectral gap and the Cheeger's constant of G, i.e., $1-\lambda_2 \geqslant \frac{1}{2\Delta n}$, given that $1-\lambda_2 \geqslant \frac{h_G^2}{2} \geqslant \frac{1}{2\Delta n}$.

3. We know from Lemma B.1 that the length of the projection of the state vector on the stepwise vectors is not too small, i.e., $|\boldsymbol{y}(u)| \geqslant \frac{k}{\Delta n}$, w.h.p.

Since Lemma B.2 holds for every time $t < T_2$, we conclude that there exists a time window $[T_1, T_2]$ such that, for every time $t \in [T_1, T_2]$ of the AVERAGING dynamics, it holds that $\operatorname{sgn}(\boldsymbol{x}^{(t)}(u) - \boldsymbol{x}^{(t+1)}(u)) = \operatorname{sgn}(\boldsymbol{c}^{(t)}(u) - \boldsymbol{c}^{(t+1)}(u))$, with high probability. Moreover, Lemma B.2 tells us that $\operatorname{sgn}(\boldsymbol{c}^{(t)}(u) - \boldsymbol{c}^{(t+1)}(u)) = \operatorname{sgn}(\boldsymbol{y}(u))$, for every $u \in V$ and for every $t \in [T_1, T_2]$. Thus, $\operatorname{sgn}(\boldsymbol{x}^{(t)}(u) - \boldsymbol{x}^{(t+1)}(u)) = \operatorname{sgn}(\boldsymbol{y}(u))$, concluding the proof. □

In Lemma 4.2, instead, we show that with some constant probability (i.e., independent from the number of nodes n) the first two "steps" of the vector $\boldsymbol{y}$ have different signs, i.e., the sign can be considered as a criterion to distinguish the first two communities.

Lemma 4.2 (Different communities, different signs)**.** *Let $G = (V, E, w)$ be a clustered k-volume regular graph with maximum weighted degree $\Delta \leqslant poly(n)$ and $N = \mathcal{O}(\sqrt{k}/\Delta)$. For each pair of nodes $u \in V_i$ and $v \in V_j$, with $i \neq j$, it holds that*

$$\mathbf{P}\left(\operatorname{sgn}(\boldsymbol{y}(u)) \neq \operatorname{sgn}(\boldsymbol{y}(v))\right) = \Omega(1).$$

Proof. Since the ordering of the communities (and consequent definition of the $\boldsymbol{\chi}_i$'s) is completely arbitrary, we can without loss of generality assume $i = 1$ and $j = 2$. From Lemma 4.1 we have that $\operatorname{sgn}(\boldsymbol{x}^{(t)}(u) - \boldsymbol{x}^{(t+1)}(u)) = \operatorname{sgn}(\boldsymbol{y}(u))$, for every $u \in V$, during a time interval $[T_1, T_2]$, w.h.p. Let us define $X(V_i) := \sum_{w \in V_i} \delta(w)\boldsymbol{x}(w)$.

Note that $\boldsymbol{y}(u) = \gamma_1 \boldsymbol{\chi}_1(u)$ and $\boldsymbol{y}(v) = \gamma_1 \boldsymbol{\chi}_1(v) + \gamma_2 \boldsymbol{\chi}_2(v)$, since the other terms of the $\boldsymbol{\chi}_i$s are equal to 0 on the components relative to u and v. Thus, with some algebra, we get

$$\boldsymbol{y}(u) = \frac{1}{m}\left[\frac{\hat{m}_1}{m_1}X(V_1) - X(V_2) - X(\hat{V}_2)\right], \boldsymbol{y}(v) = \frac{1}{m}\left[\frac{m_1 m_2 + m\hat{m}_2}{\hat{m}_1 m_2}X(V_2) - X(V_1) - X(\hat{V}_2)\right].$$

Note that, by linearity of expectation, $\mathbf{E}[X(V_i)] = 0$. Moreover, since the terms $\boldsymbol{x}(w)$s are independent Rademacher random variables, we can write the standard deviation of $X(V_i)$ as

$$\sigma(X(V_i)) = \sqrt{\sum_{w \in V_i} \sigma^2(\boldsymbol{x}(w))} = \sqrt{\sum_{w \in V_i} \left(\mathbf{E}[\delta(w)^2 \boldsymbol{x}(w)^2] - \mathbf{E}[\delta(w)\boldsymbol{x}(w)]^2\right)} = \sqrt{\sum_{w \in V_i} \delta(w)^2}.$$

Then we can upper and lower bound the standard deviation $\sigma(X(V_i))$ getting $\frac{m_i}{\sqrt{|V_i|}} \leqslant \sigma(X(V_i)) \leqslant \Delta\sqrt{|V_i|}$, where the lower bound follows from $\|\boldsymbol{d}\|_2 \geqslant \|\boldsymbol{d}\|_1 / \sqrt{|V_i|}$, where $\boldsymbol{d_i}$ is the vector of weighted degrees of nodes in community V_i, and for the upper bound we used that $\delta(w) \leqslant \Delta$, for each $w \in V$.

Let us now define the following three events:

1. E_1: $X(V_1) \geqslant \sigma(X(V_1)) \implies X(V_1) \geqslant \frac{m_1}{\sqrt{|V_1|}} \geqslant \frac{\min_i m_i}{\sqrt{\max_i |V_i|}}$;

2. E_2: $X(V_2) \leqslant -\sigma(X(V_2)) \implies X(V_2) \leqslant -\frac{m_2}{\sqrt{|V_2|}} \leqslant -\frac{\min_i m_i}{\sqrt{\max_i |V_i|}}$;

3. E_3: $0 \leqslant X(\hat{V}_2) \leqslant \varepsilon\sigma(X(\hat{V}_2)) \implies 0 \leqslant X(\hat{V}_2) \leqslant \varepsilon\Delta\sqrt{\sum_{i=3}^{k} |V_i|} \leqslant \varepsilon\Delta\sqrt{k \max_i |V_i|}$,

with ε a suitable positive constant. When E_1, E_2, E_3 are true, i.e., with some constant probability, it holds that $\boldsymbol{y}(v) < 0$; as for $\boldsymbol{y}(u)$ we have that

$$\begin{aligned}\frac{\hat{m}_1}{m_1}X(V_1) - X(V_2) - X(\hat{V}_2) &\geqslant \frac{\hat{m}_1}{m_1}\sigma(X(V_1)) + \sigma(X(V_2)) - \varepsilon\sigma(X(\hat{V}_2)) \\ &\geqslant \frac{k\min_i |V_i|}{\sqrt{\max_i |V_i|}} - \varepsilon\Delta\sqrt{k\max_i |V_i|}.\end{aligned}$$

The previous inequality is greater than 0 whenever $\varepsilon < \frac{\sqrt{k}}{\Delta N}$. By hypothesis $\Delta N = \mathcal{O}(\sqrt{k})$ and thus $\frac{\sqrt{k}}{\Delta N} = \Omega(1)$, i.e., there is an $\varepsilon = \Omega(1)$ such that $\boldsymbol{y}(u) > 0$.

By approximating the random variables with Gaussian ones and using Berry-Esseen's theorem (Theorem A.3), it is possible to show that all three events have probability at least constant; moreover, being the events independent, also $\mathbf{P}(E_1, E_2, E_3)$ is constant. □

Proof of Theorem 4.1. The proof proceeds by showing that the binary labeling of the nodes of G produced by the AVERAGING dynamics during the time window $[T_1, T_2]$ is such that the two conditions required by the definition of (ε, δ)-community sensitive algorithm (Definition 2.1) are met. The first condition follows directly from Lemma 4.1 and from the fact that $\boldsymbol{y}$ is a "stepwise" vector, with $\varepsilon = \mathcal{O}(n^{-\frac{1}{2}})$ (see Lemma B.1 for details on the probability). The second condition follows directly from Lemma 4.2. □

5 Bipartite graphs

Assume $G = (V, E, w)$ is a bipartite 2-volume regular graph, i.e., $V = V_1 \cup V_2$, $E \subseteq V_1 \times V_2$ and G is volume regular w.r.t. the bipartition (V_1, V_2). In this case, basic properties of random walks imply that the AVERAGING dynamics does not converge to the global (weighted) average of the values, but it periodically oscillates. In fact, in this case the transition matrix P has an eigenvector $\boldsymbol{\chi} = \mathbf{1}_{V_1} - \mathbf{1}_{V_2}$ with eigenvalue $\lambda_n = -1$ (as implied by Claim 5.1). Thus, the state vector is mainly affected by the eigenvectors associated to the two eigenvalues of absolute value 1 (i.e., λ_1 and λ_n). After a number of rounds of the dynamics that depends on $1/\lambda_2$, we have that, in even rounds, all nodes in V_i $(i = 1, 2)$ have a state that is close to some local average μ_i; in odd rounds, these values are swapped (as shown in Eq. (5)).

If one were observing the process in even rounds,[6] however, the states of nodes in V_1 would converge to μ_1 and those of nodes in V_2 would converge to μ_2. Unfortunately (and differently from clustered volume regular graphs), convergence to the local average for nodes belonging to the same community does not eventually become monotone (i.e., increasing or decreasing). This follows since the eigenvector associated to λ_2 is no longer stepwise in general (lumpable classes are already associated to $\mathbf{1}$ and $\boldsymbol{\chi}$). However, we can easily modify the labeling scheme of the AVERAGING dynamics to perform *bipartiteness detection* as follows: Nodes apply the labeling rule every two time steps and they do it between the states of two consecutive rounds, i.e., each node $v \in V$ sets $\texttt{label}^{(2t)}(v) = 1$ if $\boldsymbol{x}^{(2t)}(v) \geqslant \boldsymbol{x}^{(2t-1)}(v)$ and $\texttt{label}^{(2t)}(v) = 0$ otherwise. We call this new protocol AVERAGING BIPARTITE dynamics.

We now show how AVERAGING BIPARTITE dynamics can perform *bipartiteness detection*. Recall that we denote with $W \in \mathbb{R}^{n\times n}$ the weighted adjacency matrix of G. Since G is undirected and bipartite, the matrix W can be written as

$$W = \begin{pmatrix} 0 & W_1 \\ W_2 & 0 \end{pmatrix} = \begin{pmatrix} 0 & W_1 \\ W_1^T & 0 \end{pmatrix}.$$

[6] Or, equivalently, in odd rounds.

Thus, the transition matrix of a simple random walk on G, i.e., $P = D^{-1}W$ where D^{-1} is a diagonal matrix and $D_{ii} = \frac{1}{\delta(i)}$, has the form

$$P = \begin{pmatrix} 0 & P_1 \\ P_1^T & 0 \end{pmatrix}.$$

Claim 5.1 shows that the spectrum of P is symmetric and it gives a relation between the eigenvectors of symmetric eigenvalues.

Claim 5.1. *Let $G = (V_1 \cup V_2, E, w)$ be an edge-weighted undirected bipartite graph with bipartition (V_1, V_2) and such that $|V_i| = n_i$. If $\boldsymbol{v} = (\boldsymbol{v}_1, \boldsymbol{v}_2)^T$, with $\boldsymbol{v}_i \in \mathbb{R}^{n_i}$, is an eigenvector of P with eigenvalue λ, then $\boldsymbol{v}' = (\boldsymbol{v}_1, -\boldsymbol{v}_2)^T$ is an eigenvector of P with eigenvalue $-\lambda$.*

Proof. If $P\boldsymbol{v} = \lambda\boldsymbol{v}$ then we have that $P_1\boldsymbol{v}_2 = \lambda\boldsymbol{v}_1$ and $P_1^T\boldsymbol{v}_2 = \lambda\boldsymbol{v}_2$. Using these two equalities we get that $P\boldsymbol{v}' = -\lambda\boldsymbol{v}'$. In fact,

$$P\boldsymbol{v}' = \begin{pmatrix} 0 & P_1 \\ P_1^T & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{v}_1 \\ -\boldsymbol{v}_2 \end{pmatrix} = \begin{pmatrix} -P_1\boldsymbol{v}_2 \\ P_1^T\boldsymbol{v}_1 \end{pmatrix} = -\lambda \begin{pmatrix} \boldsymbol{v}_1 \\ -\boldsymbol{v}_2 \end{pmatrix}.$$

□

The transition matrix P is stochastic, thus the vector $\mathbf{1}$ (i.e., the vector of all ones) is an eigenvector associated to $\lambda_1 = 1$, that is the first largest eigenvalue of P. Claim 5.1 implies that $\boldsymbol{\chi} = \mathbf{1}_{V_1} - \mathbf{1}_{V_2}$ is an eigenvector of P with eigenvalue $\lambda_n = -1$.

As in Section 2, we write the state vector at time t using the spectral decomposition of P. Let $1 = \lambda_1 > \lambda_2 \geqslant \ldots > \lambda_n = -1$ be the eigenvalues of P. We denote by $\mathbf{1} = \boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n = \boldsymbol{\chi}$ a family of n linearly independent eigenvectors of P, where each $\boldsymbol{v}_i$ is the eigenvector associated to λ_i. Thus, we have that

$$\boldsymbol{x}^{(t)} = P^t\boldsymbol{x} = \sum_{i=1}^{n} \lambda_i^t \alpha_i \boldsymbol{v}_i = \alpha_1 \mathbf{1} + (-1)^t \alpha_n \boldsymbol{\chi} + \sum_{i=2}^{n-1} \lambda_i^t \alpha_i \boldsymbol{v}_i \tag{5}$$

where $\alpha_i = \frac{\langle D^{\frac{1}{2}}\boldsymbol{x}, D^{\frac{1}{2}}\boldsymbol{v}_i\rangle}{\|D^{\frac{1}{2}}\boldsymbol{v}_i\|^2}$. The last equation implies that $\boldsymbol{x}^{(t)} = P^t\boldsymbol{x}$ does not converge to some value as t tends to infinity, but oscillates. In particular, nodes in V_1 on even rounds and nodes in V_2 on odd rounds, converge to $\alpha_1 + \alpha_n$. Instead in the symmetric case, i.e., odd rounds for nodes in V_1 and even rounds for nodes in V_2, the process converges to $\alpha_1 - \alpha_n$. These quantities are proportional to the weighted average of the initial values in the first and in the second partition, respectively.

Lemma 5.1, whose proof follows, shows that Averaging Bipartite dynamics performs bipartiteness detection in $\mathcal{O}(\log n \,/\, \log(1/\lambda_2))$ rounds. Note that if $\log(1/\lambda_2) = \Omega(1)$, then the Averaging Bipartite dynamics takes logarithmic time to find the bipartition.

Lemma 5.1. *Let $G = (V, E, w)$ be an edge-weighted bipartite volume regular graph with bipartition V_1, V_2 and maximum weighted degree $\Delta \leqslant poly(n)$. Then for every time $t > T$, with $T = \mathcal{O}(\log n \,/\, \log(1/\lambda_2))$, the* Averaging Bipartite *dynamics is a $(\mathcal{O}(n^{-\frac{1}{2}}), \mathcal{O}(1))$-community sensitive algorithm, w.h.p.*

Proof. We assume that the coloring rule is applied between every even and every odd round (conversely, the signs of the nodes in the analysis are swapped). Recall the definition of the

error contribution, namely $\boldsymbol{e}^{(t)}(u) = \sum_{i=2}^{n-1} \lambda_i^t \alpha_i \boldsymbol{v}_i(u)$. We compute the difference between the state vectors of two consecutive steps by using Eq. (5), namely

$$\begin{aligned}\boldsymbol{x}^{(2t)} - \boldsymbol{x}^{(2t+1)} &= \alpha_1 \mathbf{1} + (-1)^{2t}\alpha_n \boldsymbol{\chi} + \boldsymbol{e}^{(2t)} - \alpha_1 \mathbf{1} - (-1)^{2t+1}\alpha_n \boldsymbol{\chi} - \boldsymbol{e}^{(2t+1)} \\ &= 2\alpha_n \boldsymbol{\chi} + \boldsymbol{e}^{(2t)} - \boldsymbol{e}^{(2t+1)}.\end{aligned}$$

We want to find a time T such that for every $t > T$ the sign of a node $u \in V$ depends only on $\boldsymbol{\chi}(u)$. Formally, $\operatorname{sgn}(\boldsymbol{x}^{(2t)}(u) - \boldsymbol{x}^{(2t+1)}(u)) = \operatorname{sgn}(\alpha_n \boldsymbol{\chi})$. The last equation holds whenever

$$\begin{aligned}2|\alpha_n \boldsymbol{\chi}(u)| &> |\boldsymbol{e}^{(2t)}(u) - \boldsymbol{e}^{(2t+1)}(u)| \\ 2|\alpha_n| &> |\boldsymbol{e}^{(2t)}(u) - \boldsymbol{e}^{(2t+1)}(u)|. \end{aligned} \tag{6}$$

We upper bound $|\boldsymbol{e}^{(2t)}(u) - \boldsymbol{e}^{(2t+1)}(u)|$ by using Lemma B.3. We get that $|\boldsymbol{e}^{(2t)}(u) - \boldsymbol{e}^{(2t+1)}(u)| \leqslant 2\lambda_2^{2t}\sqrt{\Delta n}$. We get that Eq. (6) holds if the following holds:

$$\begin{aligned}|\alpha_n| &> \lambda_2^{2t}\sqrt{\Delta n} \\ \left(\frac{1}{\lambda_2}\right)^{2t} &> \frac{\sqrt{\Delta n}}{|\alpha_n|} \\ 2t &> \log\left(\frac{\sqrt{\Delta n}}{|\alpha_n|}\right)\frac{1}{\log(1/\lambda_2)}.\end{aligned}$$

In order to find the time t which makes the last inequality hold, we provide a lower bound on $|\alpha_n|$, showing that it is not too small, with high probability. Recall that $\alpha_i = \frac{\langle D^{\frac{1}{2}}\boldsymbol{x}, D^{\frac{1}{2}}\boldsymbol{v}_i\rangle}{\|D^{\frac{1}{2}}\boldsymbol{v}_i\|^2}$ and thus

$$\alpha_n = \frac{\langle D^{\frac{1}{2}}\boldsymbol{x}, D^{\frac{1}{2}}\boldsymbol{\chi}\rangle}{\|D^{\frac{1}{2}}\boldsymbol{\chi}\|^2} = \frac{1}{\operatorname{vol}(V)}\sum_{v\in V}\delta(v)\boldsymbol{x}(v)\boldsymbol{\chi}(v),$$

where $\operatorname{vol}(V) = \sum_{v\in V}\delta(v)$. We get the lower bound, with high probability, by showing that

$$\mathbf{P}\left(|\alpha_n| \leqslant \frac{1}{\Delta n}\right) \leqslant \mathbf{P}\left(|\alpha_n| \leqslant \frac{1}{\operatorname{vol}(V)}\right) = \mathbf{P}\left(\left|\sum_{v\in V}\delta(v)\boldsymbol{x}(v)\boldsymbol{\chi}(v)\right| \leqslant 1\right) \overset{(a)}{=} \mathcal{O}\left(\frac{1}{\sqrt{n}}\right)$$

where in (a) we apply Theorem A.4. Indeed this last inequality implies that $|\alpha_n| > \frac{1}{\Delta n}$ with high probability. The thesis then follows from the above bound on $|\alpha_n|$ and from the hypothesis on $\Delta \leqslant poly(n)$. □

6 Other non-clustered volume regular graphs

Consider k-volume regular graphs whose k stepwise eigenvectors are associated to the k largest eigenvalues, in absolute value. These graphs include many k-partite graphs (e.g., regular ones), graphs that are "close" to being k-partite (i.e., ones that would become k-partite upon removal of a few edges). Differently from the clustered case (Theorem 4.1) some of the k eigenvalues can in general be negative.

Consider the following variant of the labeling scheme of the AVERAGING dynamics, in which nodes apply their labeling rule only on even rounds, comparing their value with the one they held at the end of the last even round, i.e., each node $v \in V$ sets $\texttt{label}^{(2t)}(v) = 1$ if $\boldsymbol{x}^{(2t)}(v) \geqslant \boldsymbol{x}^{(2t-2)}(v)$ and $\texttt{label}^{(2t)}(v) = 0$ otherwise. Since the above protocol amounts to only taking even powers of eigenvalues, the analysis of this modified protocol proceeds along the same lines as the clustered case, while the results of Theorem 4.1 seamlessly extend to this class of graphs.

7 Conclusions

The focus of this work is on heuristics that implicitely perform spectral graph clustering, without explicitely computing the main eigenvectors of a matrix describing connectivity properties of the underlying network (typically, its Laplacian or a related matrix). In this perspective, we extended the work of Becchetti et al. [BCN+17] in several ways. In particular, for k communities, [BCN+17] considered an extremely regular case, in which the second eigenvalue of the (normalized) Laplacian has algebraic and geometric multiplicities $k-1$ and the corresponding eigenspace is spanned by a basis of indicator vectors. We considered a more general case in which the first k eigenvalues are in general different, but the span of the corresponding eigenvectors again admits a base of indicator vectors. We also made a connection between this stepwise property and lumpability properties of the underlying random walk, which results in a class of volume-regular graphs, that may not have constant degree, nor exhibit balanced communities.

Though far from conclusive, we believe our results point to potentially interesting directions for future research. In general, our analysis sheds further light on the connections between temporal evolution of the power method and spectral-related clustering properties of the underlying network. At the same time, we showed that variants of the AVERAGING dynamics (and/or its labeling rule) might be useful in addressing different problems and/or other graph classes, as the examples given in Section 5 suggest. On the other hand, identifying k hidden partitions using the algorithm presented in [BCN+17] requires relatively strong assumptions on the k main eigenvalues and knowledge of an upper bound to the graph size,[7] while the analysis becomes considerably more intricate than the perfectly regular and completely balanced case addressed in [BCN+17]. Some aspects of our analysis (e.g., the aforementioned presence of a size-dependent time window in which the labeling rule has to be applied) suggest that more sophisticated variants of the AVERAGING dynamics might be needed to express the full power of a spectral method that explicitely computes the k main eigenvectors of a graph-related matrix. While we believe this goal can be achieved, designing and analyzing such an algorithm might prove a challenging task.

References

[AAE08] Dana Angluin, James Aspnes, and David Eisenstat. A Simple Population Protocol for Fast Robust Approximate Majority. *Distributed Computing*, 21(2):87–102, 2008. (Preliminary version in DISC'07).

[ABH14] Emmanuel Abbe, Afonso S. Bandeira, and Georgina Hall. Exact recovery in the stochastic block model. *IEEE Trans. on Information Theory*, 62(1):471–487, 2014.

[AS15] Emmanuel Abbe and Colin Sandon. Detection in the stochastic block model with multiple clusters: proof of the achievability conjectures, acyclic bp, and the information-computation gap. *arXiv preprint arXiv:1512.09080*, 2015.

[BC09] Michael J. Barber and John W. Clark. Detecting network communities by propagating labels under constraints. *Physical Review E*, 80(2):026129, 2009.

[BCM+18] Luca Becchetti, Andrea E.F. Clementi, Pasin Manurangsi, Emanuele Natale, Francesco Pasquale, Prasad Raghavendra, and Luca Trevisan. Average whenever

[7] As anecdotal experimental evidence suggests, the presence of a time window to perform labeling is not an artifact of our analysis.

you meet: Opportunistic protocols for community detection. In *26th Annual European Symposium on Algorithms, ESA 2018, August 20-22, 2018, Helsinki, Finland*, pages 7:1–7:13, 2018.

[BCN+17] Luca Becchetti, Andrea E.F. Clementi, Emanuele Natale, Francesco Pasquale, and Luca Trevisan. Find your place: Simple distributed algorithms for community detection. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2017, Barcelona, Spain, Hotel Porta Fira, January 16-19*, pages 940–959, 2017.

[BLM15] Charles Bordenave, Marc Lelarge, and Laurent Massoulié. Non-backtracking spectrum of random graphs: community detection and non-regular ramanujan graphs. In *Proc. of 56rd Ann. IEEE Symp. on Foundations of Computer Science (FOCS'15)*, pages 1347–1357. IEEE, 2015.

[Bop87] Ravi B. Boppana. Eigenvalues and graph bisection: An average-case analysis. In *Proc. of the 28th Ann. IEEE Symp. on Foundations of Computer Science (FOCS'87)*, pages 280–285. IEEE, 1987.

[Chu96] Fan R.K. Chung. Laplacians of graphs and Cheeger's inequalities. In *Combinatorics, Paul Erdös is eighty*, volume 2, pages 157–172. János Bolyai Math. Soc., 1996.

[CNNS18] Emilio Cruciani, Emanuele Natale, André Nusser, and Giacomo Scornavacca. Phase transition of the 2-Choices dynamics on core-periphery networks. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2018, Stockholm, Sweden, July 10-15, 2018*, pages 777–785, 2018.

[CNS19] Emilio Cruciani, Emanuele Natale, and Giacomo Scornavacca. Distributed community detection via metastability of the 2-Choices dynamics. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence, AAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, 2019.

[CO05] Amin Coja-Oghlan. *Spectral techniques, semidefinite programs, and random graphs.* Habilitation thesis, Humboldt University Berlin, 2005.

[CO10] Amin Coja-Oghlan. Graph partitioning via adaptive spectral techniques. *Combinatorics, Probability and Computing*, 19(02):227–284, 2010.

[DF89] Martin E. Dyer and Alan M. Frieze. The solution of some random NP-hard problems in polynomial expected time. *Journal of Algorithms*, 10(4):451–489, 1989.

[DKMZ11] Aurelien Decelle, Florent Krzakala, Cristopher Moore, and Lenka Zdeborová. Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications. *Physical Review E*, 84(6):066106, 2011.

[Erd45] Paul Erdös. On a lemma of littlewood and offord. *Bulletin of the American Mathematical Society*, 51(12):898–902, 1945.

[Fie89] Miroslav Fiedler. Laplacian of graphs and algebraic connectivity. *Banach Center Publications*, 25(1):57–70, 1989.

[For10] Santo Fortunato. Community detection in graphs. *Physics Reports*, 486(3):75–174, 2010.

[HLL83] Paul W. Holland, Kathryn Blackmond Laskey, and Samuel Leinhardt. Stochastic blockmodels: First steps. *Social networks*, 5(2):109–137, 1983.

[HP01] Yehuda Hassin and David Peleg. Distributed probabilistic polling and applications to proportionate agreement. *Information and Computation*, 171(2):248–268, 2001.

[JS98] Mark Jerrum and Gregory B. Sorkin. The metropolis algorithm for graph bisection. *Discrete Applied Mathematics*, 82(1):155–175, 1998.

[KM04] David Kempe and Frank McSherry. A decentralized algorithm for spectral analysis. In *Proc. of the 36th Ann. ACM Symp. on Theory of Computing (STOC'04)*, pages 561–568, 2004.

[KMM+13] Florent Krzakala, Cristopher Moore, Elchanan Mossel, Joe Neeman, Allan Sly, Lenka Zdeborová, and Pan Zhang. Spectral redemption in clustering sparse networks. *Proceedings of the National Academy of Sciences*, 110(52):20935–20940, 2013.

[KPS13] Kishore Kothapalli, Sriram V. Pemmaraju, and Vivek Sardeshmukh. On the analysis of a label propagation algorithm for community detection. In *Proc. of the 14th Int. Conf. on Distributed Computing and Networking (ICDCN'13)*, pages 255–269, 2013.

[KS60] John G. Kemeny and J. Laurie Snell. *Finite Markov chains*. D. van Nostrand Company, inc., Princeton, N.J., 1960.

[Lan50] Cornelius Lanczos. *An iteration method for the solution of the eigenvalue problem of linear differential and integral operators*. United States Governm. Press Office Los Angeles, CA, 1950.

[LGT14] James R Lee, Shayan Oveis Gharan, and Luca Trevisan. Multiway spectral partitioning and higher-order cheeger inequalities. *Journal of the ACM (JACM)*, 61(6):37, 2014.

[LM10] Xin Liu and Tsuyoshi Murata. Advanced modularity-specialized label propagation algorithm for detecting communities in networks. *Physica A: Statistical Mechanics and its Applications*, 389(7):1493–1500, 2010.

[LRU14] Jure Leskovec, Anand Rajaraman, and Jeffrey David Ullman. *Mining of massive datasets*. Cambridge university press, 2014.

[Mac03] David J.C. MacKay. *Information theory, inference and learning algorithms*. Cambridge university press, 2003.

[McS01] Frank McSherry. Spectral partitioning of random graphs. In *Proc. of the 42nd IEEE Symp. on Foundations of Computer Science (FOCS'01)*, pages 529–537, 2001.

[MK07] Joris M. Mooij and Hilbert J. Kappen. Sufficient conditions for convergence of the sum–product algorithm. *IEEE Transactions on Information Theory*, 53(12):4422–4437, 2007.

[MMM18] Frederik Mallmann-Trenn, Cameron Musco, and Christopher Musco. Eigenvector computation and community detection in asynchronous gossip models. In *45th International Colloquium on Automata, Languages, and Programming, ICALP 2018, July 9-13, 2018, Prague, Czech Republic*, pages 159:1–159:14, 2018.

[MNS13] Elchanan Mossel, Joe Neeman, and Allan Sly. A proof of the block model threshold conjecture. *arXiv preprint arXiv:1311.4115*, 2013.

[MNS16] Elchanan Mossel, Joe Neeman, and Allan Sly. Belief propagation, robust reconstruction and optimal recovery of block models. *Ann. Appl. Probab.*, 26(4):2211–2256, 08 2016.

[MS90] Stephen J Montgomery-Smith. The distribution of rademacher sums. *Proceedings of the American Mathematical Society*, 109(2):517–522, 1990.

[NJW02] Andrew Y Ng, Michael I Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in neural information processing systems*, pages 849–856, 2002.

[PSZ17] Richard Peng, He Sun, and Luca Zanetti. Partitioning well-clustered graphs: Spectral clustering works! *SIAM Journal on Computing*, 46(2):710–743, 2017.

[RAK07] Usha Nandini Raghavan, Réka Albert, and Soundar Kumara. Near linear time algorithm to detect community structures in large-scale networks. *Physical Review E*, 76(3):036106, 2007.

[She14] Irina Shevtsova. On the absolute constants in the berry-esseen-type inequalities. *Doklady Mathematics*, 89(3):378–381, May 2014.

[SM00] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *Departmental Papers (CIS)*, page 107, 2000.

[SZ17] He Sun and Luca Zanetti. Distributed graph clustering and sparsification. *CoRR*, abs/1711.01262, 2017.

[TK06] Jianjun Paul Tian and D Kannan. Lumpability and commutativity of markov processes. *Stochastic analysis and Applications*, 24(3):685–702, 2006.

[VL07] Ulrike Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416, 2007.

[Wei00] Yair Weiss. Correctness of local probability propagation in graphical models with loops. *Neural computation*, 12(1):1–41, 2000.

Appendix

A Useful inequalities

Theorem A.1 (Cauchy-Schwarz's inequality)**.** *For all vectors $\boldsymbol{u}, \boldsymbol{v}$ of an inner product space it holds that $\langle \boldsymbol{u}, \boldsymbol{v} \rangle \leqslant \langle \boldsymbol{u}, \boldsymbol{u} \rangle \cdot \langle \boldsymbol{v}, \boldsymbol{v} \rangle$, where $\langle \cdot, \cdot \rangle$ is the inner product.*

Theorem A.2 (Cheeger's inequality [Chu96])**.** *Let P be the transition matrix of a connected edge-weighted graph $G = (V, E, w)$ and let λ_2 be its second largest eigenvalue. Let $|E(S, V \setminus S)| = \sum_{u \in S,\, v \in V \setminus S} w(u, v)$ and $h_G = \min_{S : \mathrm{vol}(S) \leqslant \frac{\mathrm{vol}(V)}{2}} \frac{|E(S, V \setminus S)|}{\mathrm{vol}(S)}$. Then*

$$\frac{1 - \lambda_2}{2} \leqslant h_G \leqslant \sqrt{2(1 - \lambda_2)}.$$

Theorem A.3 (Berry-Esseen's theorem)**.** *Let $X_1, \ldots, X_n$ be independent and identically distributed random variables with mean $\mu = 0$, variance $\sigma^2 > 0$, and third absolute moment $\rho < \infty$. Let $Y_n = \frac{1}{n} \sum_{i=1}^{n} X_i$; let F_n be the cumulative distribution function of $\frac{Y_n \sqrt{n}}{\sigma}$; let Φ the cumulative distribution function of the standard normal distribution. Then, there exists a positive constant $C < 0.4748$ ([She14]) such that, for all x and for all n,*

$$|F_n(x) - \Phi(x)| \leqslant \frac{C \rho}{\sigma^3 \sqrt{n}}.$$

Theorem A.4 (Littlewood-Offord's small ball [Erd45])**.** *Let x_i be a Rademacher random variable (taking values ± 1 with probability $p = \frac{1}{2}$), let a_i be real constants such that $|a_i| \geqslant 1$, and let $X = \sum_{i=1}^{n} a_i x_i$. Then, for any $r \in \mathbb{R}$, it holds that*

$$\mathbf{P}(|X - r| < 1) = \mathcal{O}\left(\frac{1}{\sqrt{n}}\right).$$

Theorem A.5 (Rademacher concentration bound [MS90])**.** *Let x_i be a Rademacher random variable (taking values ± 1 with probability $p = \frac{1}{2}$), let a_i be real constants, and let $X = \sum_{i=1}^{n} a_i x_i$. Then, it holds that*

$$\mathbf{P}\left(|X| > t \|\boldsymbol{a}\|_2\right) \leqslant 2e^{-\frac{t^2}{2}}.$$

where $\|\boldsymbol{a}\|_2$ is the Euclidean norm of the vector $\boldsymbol{a} = (a_1, \ldots, a_n)$.

B Proofs for Lemma 4.1

In this section we prove all technical lemmas used in the proof of Lemma 4.1. The main result in Appendix B.1 is Lemma B.1 in which we show that every component of $\boldsymbol{y}$, i.e., the projection of the (inhomogeneously) contracted initial state vector $D^{\frac{1}{2}} \boldsymbol{x}$ on the (inhomogeneously) contracted vectors $D^{\frac{1}{2}} \boldsymbol{\chi}_i$s, is not too small, w.h.p. This result is shown first since it is used in other lemmas in this section. In Appendix B.2 we provide a lower bound on the core contribution of the state vector between two consecutive time steps. Moreover, we show that the sign of a node depends only on the sign of $\boldsymbol{y}$. Finally, in Appendix B.3 we upper bound the error (i.e., the part of the state vector in the eigenspace of eigenvalue $\lambda_{k+1}, \ldots, \lambda_n$).

B.1 Length of the projection of the state vector

Claim B.1. *Let $\alpha(u,v) = \sum_{i=1}^{k-1} \frac{\boldsymbol{\chi}_i(u)\boldsymbol{\chi}_i(v)}{\hat{m}_{i-1}}$. For every pair of nodes $u, v \in V$ it holds that*

$$\min_{u,v\in V} |\alpha(u,v)| \geqslant \frac{k}{\Delta n}.$$

Proof. Let $u \in V_l$ and $v \in V_h$, for some $l, h \in [k]$. We divide the proof in two cases. First, we assume that $l = h$, then we handle the case $l \neq h$. Without loss of generality, we assume $m_1 \leqslant \ldots \leqslant m_k$ and consequently $m = \hat{m}_0 \geqslant \hat{m}_1 \geqslant \ldots \geqslant \hat{m}_{k-1} = m_k$.

- Let us suppose $l = h$. Then

$$\begin{aligned}\min_{u,v\in V} |\alpha(u,v)| &= \min_{u,v\in V} \left(\sum_{i<\min\{h,l\}} \frac{m_i}{\hat{m}_i\hat{m}_{i-1}} + \frac{\hat{m}_l}{m_l\hat{m}_{l-1}} \right) \geqslant \frac{\hat{m}_1}{m_1\hat{m}_0} \\ &= \frac{m_2 + \ldots + m_k}{m_1 m} \geqslant \frac{k}{m} \geqslant \frac{k}{n \max_v \delta(v)} \geqslant \frac{k}{\Delta n}.\end{aligned}$$

- Let us suppose $l \neq h$. Note that, in this case, $\alpha(u,v) = \sum_{i<\min\{h,l\}} \frac{m_i}{\hat{m}_i\hat{m}_{i-1}} - 1 < 0$. In fact,

$$\begin{aligned}\sum_{i<\min\{h,l\}} \frac{m_i}{\hat{m}_i\hat{m}_{i-1}} &= \sum_{i<\min\{h,l\}} \frac{m_i}{\left(\sum_{j=i+1}^{k} m_j\right)\left(\sum_{j=i}^{k} m_j\right)} \\ &\overset{(a)}{\leqslant} \sum_{i<\min\{h,l\}} \frac{m_i}{(k-i)(k-i+1)m_i^2} = \sum_{i<\min\{h,l\}} \frac{1}{(k-i)(k-i+1)m_i} \\ &\leqslant \sum_{i<\min\{h,l\}} \frac{1}{(k-i)(k-i+1)} \leqslant \sum_{j=1}^{k} \frac{1}{j(j+1)} \\ &= \sum_{j=1}^{k-1} \frac{1}{j} - \sum_{j=1}^{k-1} \frac{1}{j+1} = 1 - \frac{1}{k} < 1,\end{aligned}$$

  where in (a) we use the assumption on the ordering of the volumes of the communities, i.e., $m_i \leqslant m_j$ for every $i \leqslant j$. Since $\alpha(u,v) < 0$, we have that

$$|\alpha(u,v)| = 1 - \sum_{i<\min\{h,l\}} \frac{m_i}{\hat{m}_i\hat{m}_{i-1}}.$$

  Thus,

$$\min_{u,v\in V} |\alpha(u,v)| = 1 - \max_{u,v\in V} \left(\sum_{i<\min\{h,l\}} \frac{m_i}{\hat{m}_i\hat{m}_{i-1}} \right) \geqslant 1 - \frac{(k-2)m_k}{m_k^2} = 1 - \frac{k-2}{m_k}.$$

  Note that $m_k \geqslant \frac{n}{k}$ and, given that $k \leqslant \sqrt{n}$, we get $m_k \geqslant k$. Thus

$$1 - \frac{k-2}{m_k} \geqslant \frac{2}{k} \geqslant \frac{k}{\Delta n}.$$

□

Lemma B.1 (Length of the projection of the state vector)**.** *For every $u \in V$, it holds that*

$$\mathbf{P}\left(|\boldsymbol{y}(u)| \geqslant \frac{k}{\Delta n}\right) \geqslant 1 - \mathcal{O}\left(\frac{1}{\sqrt{n}}\right).$$

Proof. Let us write $\boldsymbol{y}(u) = \sum_{i=1}^{k-1} \gamma_i \boldsymbol{\chi}_i(u)$ in terms of $\boldsymbol{x}$. Recall that

$$\gamma_i = \frac{\boldsymbol{x}^\intercal D \boldsymbol{\chi}_i}{\left\|D^{1/2}\boldsymbol{\chi}_i\right\|^2}, \quad \boldsymbol{\chi}_i = \sqrt{\frac{\hat{m}_i}{m_i}}\mathbf{1}_{V_i} - \sqrt{\frac{m_i}{\hat{m}_i}}\mathbf{1}_{\hat{V}_i}.$$

Thus, we get

$$\left\|D^{1/2}\boldsymbol{\chi}_i\right\|^2 = \frac{\hat{m}_i}{m_i}\sum_{v\in V_i}\delta(v) + \frac{m_i}{\hat{m}_i}\sum_{v\in \hat{V}_i}\delta(v) = \hat{m}_i + m_i = \hat{m}_{i-1},$$

where $\hat{m}_0 := m = \sum_{v\in V}\delta(v)$. Now, we can rewrite $\boldsymbol{y}(u)$ as

$$\begin{aligned}\boldsymbol{y}(u) &= \sum_{i=1}^{k-1}\gamma_i\boldsymbol{\chi}_i(u) = \sum_{i=1}^{k-1}\frac{\boldsymbol{x}^\intercal D\boldsymbol{\chi}_i}{\hat{m}_{i-1}}\boldsymbol{\chi}_i(u) = \sum_{i=1}^{k-1}\left(\sum_{v\in V}\frac{\delta(v)\boldsymbol{x}(v)\boldsymbol{\chi}_i(v)}{\hat{m}_{i-1}}\right)\boldsymbol{\chi}_i(u)\\ &= \sum_{v\in V}\left(\sum_{i=1}^{k-1}\frac{\boldsymbol{\chi}_i(u)\boldsymbol{\chi}_i(v)}{\hat{m}_{i-1}}\right)\delta(v)\boldsymbol{x}(v) = \sum_{v\in V}\alpha(u,v)\delta(v)\boldsymbol{x}(v),\end{aligned}$$

where

$$\alpha(u,v) := \sum_{i=1}^{k-1}\frac{\boldsymbol{\chi}_i(u)\boldsymbol{\chi}_i(v)}{\hat{m}_{i-1}}.$$

Note that, for every $u \in V_l$ and $v \in V_h$, with $l, h \in [k]$, we have

$$\boldsymbol{\chi}_i(u)\boldsymbol{\chi}_i(v) = \begin{cases} \frac{m_i}{\hat{m}_i} & \text{if } i < \min(l,h)\\ \frac{\hat{m}_i}{m_i} & \text{if } i = \min(l,h) \text{ and } l = h\\ -1 & \text{if } i = \min(l,h) \text{ and } l \neq h\\ 0 & \text{if } i > \min(l,h).\end{cases}$$

Thus, $\alpha(u,v)$ is equal to

$$\alpha(u,v) = \sum_{i=1}^{k-1}\frac{\boldsymbol{\chi}_i(u)\boldsymbol{\chi}_i(v)}{\hat{m}_{i-1}} = \begin{cases} \displaystyle\sum_{i<\min\{h,l\}}\frac{m_i}{\hat{m}_i\hat{m}_{i-1}} + \frac{\hat{m}_l}{m_l\hat{m}_{l-1}} & \text{if } h = l,\\ \displaystyle\sum_{i<\min\{h,l\}}\frac{m_i}{\hat{m}_i\hat{m}_{i-1}} - 1 & \text{if } h \neq l.\end{cases}$$

We apply Theorem A.4 and Claim B.1 to prove that the length of the projection of the state vector $\boldsymbol{x}$ on $\{\boldsymbol{\chi}_i : i \in [k]\}$ is not too small, w.h.p.

$$\begin{aligned}&\mathbf{P}\left(|\boldsymbol{y}(u)| \leqslant \frac{k}{\Delta n}\right) = \mathbf{P}\left(\left|\sum_{v\in V}\alpha(u,v)\delta(v)\boldsymbol{x}(v)\right| \leqslant \frac{k}{\Delta n}\right)\\ &= \mathbf{P}\left(\left|\sum_{v\in V}\frac{\alpha(u,v)}{\min_{u,v}|\alpha(u,v)|}\delta(v)\boldsymbol{x}(v)\right| \leqslant \frac{k}{\Delta n\min_{u,v}|\alpha(u,v)|}\right)\\ &\overset{(a)}{\leqslant} \mathbf{P}\left(\left|\sum_{v\in V}\frac{\alpha(u,v)}{\min_{u,v}|\alpha(u,v)|}\delta(v)\boldsymbol{x}(v)\right| \leqslant 1\right) \overset{(b)}{\leqslant} \mathcal{O}\left(\frac{1}{\sqrt{n}}\right),\end{aligned}$$

where in (a) we use Claim B.1 to upper bound with 1 the r.h.s. term in the probability; in (b) we can apply Theorem A.4 given that $\min_v \delta(v) = 1$ and that $\left|\frac{\alpha(u,v)}{\min_{u,v}|\alpha(u,v)|}\right| \geqslant 1$. ∎

B.2 Lower bound on the core contribution

In order to prove the main result of this section (Lemma B.2) we first provide upper and lower bounds on $\boldsymbol{c}^{(t)}(u)$ in Claim B.2; then, in Claim B.3, we use the result of Claim B.2 to bound $\boldsymbol{c}^{(t)}(u) - \boldsymbol{c}^{(t+1)}(u)$.

Claim B.2. *Let $\boldsymbol{c}^{(t)} = \sum_{i=2}^{k} \lambda_i^t \alpha_i \boldsymbol{v}_i$. For every $u \in V$ it holds that*

$$\boldsymbol{c}^{(t)}(u) \geqslant \lambda_k^t \sum_{i=2}^{k} \alpha_i \boldsymbol{v}_i(u) + t\lambda_2^{t-1}(\lambda_2 - \lambda_k) \sum_{i:\alpha_i \boldsymbol{v}_i(u)<0} \alpha_i \boldsymbol{v}_i(u),$$

$$\boldsymbol{c}^{(t)}(u) \leqslant \lambda_k^t \sum_{i=2}^{k} \alpha_i \boldsymbol{v}_i(u) + t\lambda_2^{t-1}(\lambda_2 - \lambda_k) \sum_{i:\alpha_i \boldsymbol{v}_i(u)>0} \alpha_i \boldsymbol{v}_i(u).$$

Proof. Let us start with the lower bound.

$$\begin{aligned}
\boldsymbol{c}^{(t)}(u) &= \sum_{i=2}^{k} \lambda_i^t \alpha_i \boldsymbol{v}_i(u) = \sum_{i:\alpha_i \boldsymbol{v}_i(u)>0} \lambda_i^t \alpha_i \boldsymbol{v}_i(u) + \sum_{i:\alpha_i \boldsymbol{v}_i(u)<0} \lambda_i^t \alpha_i \boldsymbol{v}_i(u) \\
&\geqslant \lambda_k \sum_{i:\alpha_i \boldsymbol{v}_i(u)>0} \lambda_i^{t-1} \alpha_i \boldsymbol{v}_i(u) + \lambda_2 \sum_{i:\alpha_i \boldsymbol{v}_i(u)<0} \lambda_i^{t-1} \alpha_i \boldsymbol{v}_i(u) \\
&\overset{(a)}{=} \lambda_k \sum_{i=2}^{k} \lambda_i^{t-1} \alpha_i \boldsymbol{v}_i(u) + (\lambda_2 - \lambda_k) \sum_{i:\alpha_i \boldsymbol{v}_i(u)<0} \lambda_i^{t-1} \alpha_i \boldsymbol{v}_i(u) \\
&\overset{(b)}{=} \lambda_k \left[\lambda_k \sum_{i=2}^{k} \lambda_i^{t-2} \alpha_i \boldsymbol{v}_i(u) + (\lambda_2 - \lambda_k) \sum_{i:\alpha_i \boldsymbol{v}_i(u)<0} \lambda_i^{t-2} \alpha_i \boldsymbol{v}_i(u) \right] \\
&\quad + (\lambda_2 - \lambda_k)\lambda_2^{t-1} \sum_{i:\alpha_i \boldsymbol{v}_i(u)<0} \alpha_i \boldsymbol{v}_i(u) \\
&= \lambda_k^2 \sum_{i=2}^{k} \lambda_i^{t-2} \alpha_i \boldsymbol{v}_i(u) + \lambda_k(\lambda_2 - \lambda_k) \sum_{i:\alpha_i \boldsymbol{v}_i(u)<0} \lambda_i^{t-2} \alpha_i \boldsymbol{v}_i(u) \\
&\quad + (\lambda_2 - \lambda_k)\lambda_2^{t-1} \sum_{i:\alpha_i \boldsymbol{v}_i(u)<0} \alpha_i \boldsymbol{v}_i(u) \\
&\geqslant \lambda_k^2 \sum_{i=2}^{k} \lambda_i^{t-2} \alpha_i \boldsymbol{v}_i(u) + \lambda_k \lambda_2^{t-2}(\lambda_2 - \lambda_k) \sum_{i:\alpha_i \boldsymbol{v}_i(u)<0} \alpha_i \boldsymbol{v}_i(u) \\
&\quad + (\lambda_2 - \lambda_k)\lambda_2^{t-1} \sum_{i:\alpha_i \boldsymbol{v}_i(u)<0} \alpha_i \boldsymbol{v}_i(u) \\
&= \lambda_k^2 \sum_{i=2}^{k} \lambda_i^{t-2} \alpha_i \boldsymbol{v}_i(u) + (\lambda_k \lambda_2^{t-2} + \lambda_2^{t-1})(\lambda_2 - \lambda_k) \sum_{i:\alpha_i \boldsymbol{v}_i(u)<0} \alpha_i \boldsymbol{v}_i(u) \\
&\geqslant \lambda_k^2 \sum_{i=2}^{k} \lambda_i^{t-2} \alpha_i \boldsymbol{v}_i(u) + 2\lambda_2^{t-1}(\lambda_2 - \lambda_k) \sum_{i:\alpha_i \boldsymbol{v}_i(u)<0} \alpha_i \boldsymbol{v}_i(u) \\
&\geqslant \dots \\
&\geqslant \lambda_k^t \sum_{i=2}^{k} \alpha_i \boldsymbol{v}_i(u) + t\lambda_2^{t-1}(\lambda_2 - \lambda_k) \sum_{i:\alpha_i \boldsymbol{v}_i(u)<0} \alpha_i \boldsymbol{v}_i(u),
\end{aligned}$$

where in (a) we add and subtract $\lambda_k \sum_{i:\alpha_i \boldsymbol{v}_i(u)<0} \lambda_i^{t-1} \alpha_i \boldsymbol{v}_i(u)$; in (b) we iterate the same reasoning on the first term only.

As for the upper bound, similarly to the previous case, we get

$$\boldsymbol{c}^{(t)}(u) \leqslant \lambda_k^t \sum_{i=2}^{k} \alpha_i \boldsymbol{v}_i(u) + t\lambda_2^{t-1}(\lambda_2 - \lambda_k) \sum_{i:\alpha_i \boldsymbol{v}_i(u)>0} \alpha_i \boldsymbol{v}_i(u).$$

□

Here we use the result of Claim B.2 to give upper and lower bounds on the difference between the *core contribution* in two consecutive rounds.

Claim B.3. *Let $\boldsymbol{c}^{(t)} = \sum_{i=2}^{k} \lambda_i^t \alpha_i \boldsymbol{v}_i$ and let $\lambda_2 > \lambda_k > \frac{1}{2}$. For every $u \in V$, it holds that*

$$\boldsymbol{c}^{(t)}(u) - \boldsymbol{c}^{(t+1)}(u) \geqslant \lambda_k^t(1-\lambda_2) \sum_{i=2}^{k} \alpha_i \boldsymbol{v}_i(u) + (t+1)\lambda_2^t(\lambda_2 - \lambda_k) \sum_{i:\alpha_i \boldsymbol{v}_i(u)<0} \alpha_i \boldsymbol{v}_i(u),$$

$$\boldsymbol{c}^{(t)}(u) - \boldsymbol{c}^{(t+1)}(u) \leqslant \lambda_k^t(1-\lambda_2) \sum_{i=2}^{k} \alpha_i \boldsymbol{v}_i(u) + (t+1)\lambda_2^t(\lambda_2 - \lambda_k) \sum_{i:\alpha_i \boldsymbol{v}_i(u)>0} \alpha_i \boldsymbol{v}_i(u).$$

Proof. Let us start with the lower bound.

$$\begin{aligned}
&\boldsymbol{c}^{(t)}(u) - \boldsymbol{c}^{(t+1)}(u) = \sum_{i=2}^{k} \lambda_i^t(1-\lambda_i)\alpha_i \boldsymbol{v}_i(u) \\
&= \sum_{i:\alpha_i \boldsymbol{v}_i(u)>0} \lambda_i^t(1-\lambda_i)\alpha_i \boldsymbol{v}_i(u) + \sum_{i:\alpha_i \boldsymbol{v}_i(u)<0} \lambda_i^t(1-\lambda_i)\alpha_i \boldsymbol{v}_i(u) \\
&\geqslant (1-\lambda_2) \sum_{i:\alpha_i \boldsymbol{v}_i(u)>0} \lambda_i^t \alpha_i \boldsymbol{v}_i(u) + (1-\lambda_k) \sum_{i:\alpha_i \boldsymbol{v}_i(u)<0} \lambda_i^t \alpha_i \boldsymbol{v}_i(u) \\
&= (1-\lambda_2) \sum_{i=2}^{k} \lambda_i^t \alpha_i \boldsymbol{v}_i(u) + (\lambda_2 - \lambda_k) \sum_{i:\alpha_i \boldsymbol{v}_i(u)<0} \lambda_i^t \alpha_i \boldsymbol{v}_i(u) \\
&\overset{(a)}{\geqslant} (1-\lambda_2) \left[\lambda_k^t \sum_{i=2}^{k} \alpha_i \boldsymbol{v}_i(u) + t\lambda_2^{t-1}(\lambda_2 - \lambda_k) \sum_{i:\alpha_i \boldsymbol{v}_i(u)<0} \alpha_i \boldsymbol{v}_i(u) \right] \\
&\qquad + \lambda_2^t(\lambda_2 - \lambda_k) \sum_{i:\alpha_i \boldsymbol{v}_i(u)<0} \alpha_i \boldsymbol{v}_i(u) \\
&= \lambda_k^t(1-\lambda_2) \sum_{i=2}^{k} \alpha_i \boldsymbol{v}_i(u) + (\lambda_2 - \lambda_k)[\lambda_2^{t-1}(1-\lambda_2)t + \lambda_2^t)] \sum_{i:\alpha_i \boldsymbol{v}_i(u)<0} \alpha_i \boldsymbol{v}_i(u) \\
&\overset{(b)}{\geqslant} \lambda_k^t(1-\lambda_2) \sum_{i=2}^{k} \alpha_i \boldsymbol{v}_i(u) + (t+1)\lambda_2^t(\lambda_2 - \lambda_k) \sum_{i:\alpha_i \boldsymbol{v}_i(u)<0} \alpha_i \boldsymbol{v}_i(u)
\end{aligned}$$

where in (a) we use Claim B.2 and $\lambda_i^t \leqslant \lambda_2^t$; in (b) we use the hypothesis on λ_2.

As for the upper bound, similarly to the previous case, we get

$$\boldsymbol{c}^{(t)}(u) - \boldsymbol{c}^{(t+1)}(u) \leqslant \lambda_k^t(1-\lambda_2) \sum_{i=2}^{k} \alpha_i \boldsymbol{v}_i(u) + (t+1)\lambda_2^t(\lambda_2 - \lambda_k) \sum_{i:\alpha_i \boldsymbol{v}_i(u)>0} \alpha_i \boldsymbol{v}_i(u).$$

□

The proof of Lemma B.2 requires one extra claim about the coefficients β_i, i.e., the ones such that $\alpha_i \boldsymbol{v}_i = \beta_i D^{\frac{1}{2}} \boldsymbol{w}_i$. This last bound is shown in Claim B.4.

Claim B.4. *Let $\boldsymbol{x} \in \{-1, 1\}^n$ be a Rademacher random vector. Let $D \in \mathbb{R}^{n \times n}$ be a positive diagonal matrix with maximum element $\Delta = \max_i D_{ii}$ and let $\boldsymbol{w} \in \mathbb{R}^n$ be a vector such that $\|\boldsymbol{w}\|_2 = 1$. Let $\beta = \langle \boldsymbol{x}, D^{\frac{1}{2}} \boldsymbol{w} \rangle$. It holds that $|\beta| \leqslant \sqrt{\Delta \log n}$, with high probability.*

Proof. Note that β is a weighted sum of Rademacher random variables with i-th coefficient equal to $(D^{\frac{1}{2}} \boldsymbol{w})(i)$ and that $\|D^{\frac{1}{2}} \boldsymbol{w}\|_2 = \sqrt{\sum_{i=1}^n \delta(i) \boldsymbol{w}(i)^2} \leqslant \sqrt{\Delta}$, since by hypothesis $\|\boldsymbol{w}\|_2 = 1$ and thus $\|D^{\frac{1}{2}} \boldsymbol{w}\|_2^2$ is a convex combination of the diagonal elements of D. Let $t = \sqrt{\log n}$; by applying Theorem A.5 we get

$$\mathbf{P}\left(|\beta| > \sqrt{\Delta \log n}\right) \leqslant \mathbf{P}\left(|\beta_i| > t \|D^{\frac{1}{2}} \boldsymbol{w}\|_2\right) \leqslant 2e^{-\frac{\log n}{2}} = \mathcal{O}\left(\frac{1}{n}\right).$$

Thus $|\beta| \leqslant \sqrt{\Delta \log n}$, with high probability. □

We now state and prove Lemma B.2.

Lemma B.2 (Lower bound on the core contribution)**.** *Let $\boldsymbol{c}^{(t)} = \sum_{i=2}^k \lambda_i^t \alpha_i \boldsymbol{v}_i$. Let $\lambda_k > \frac{1}{2}$ and $\frac{1-\lambda_2}{\lambda_2 - \lambda_k} \geqslant \Delta^{\frac{3}{2}} n^{1+c}$, for some positive constant c. For every $u \in V$ and for every time $t < T_2$, such that $T_2 = \Omega(n^{c/3})$, the two following conditions hold, w.h.p.:*

- $\left|\boldsymbol{c}^{(t)}(u) - \boldsymbol{c}^{(t+1)}(u)\right| \geqslant \frac{1}{2}\lambda_k^t (1 - \lambda_2) \left|\boldsymbol{y}(u)\right|$;
- $\operatorname{sgn}(\boldsymbol{c}^{(t)}(u) - \boldsymbol{c}^{(t+1)}(u)) = \operatorname{sgn}(\boldsymbol{y}(u))$.

Proof. We show the lower bound in the time window. To do that, first we suppose that $\boldsymbol{c}^{(t)}(u) - \boldsymbol{c}^{(t+1)}(u) > 0$ and show that the claim holds; then we show that the claim also holds when $\boldsymbol{c}^{(t)}(u) - \boldsymbol{c}^{(t+1)}(u) < 0$.

Let us suppose $\boldsymbol{c}^{(t)}(u) - \boldsymbol{c}^{(t+1)}(u) > 0$. If $\boldsymbol{y}(u) < 0$ the thesis follows directly; then let us suppose $\boldsymbol{y}(u) \geqslant 0$. From Claim B.3 we have that

$$\boldsymbol{c}^{(t)}(u) - \boldsymbol{c}^{(t+1)}(u) \geqslant \lambda_k^t (1 - \lambda_2) \sum_{i=2}^k \alpha_i \boldsymbol{v}_i(u) + (t+1)\lambda_2^t (\lambda_2 - \lambda_k) \sum_{i : \alpha_i \boldsymbol{v}_i(u) < 0} \alpha_i \boldsymbol{v}_i(u).$$

In order to prove the lemma in this first case, we need to show that

$$\frac{1}{2}\lambda_k^t (1 - \lambda_2) \sum_{i=2}^k \alpha_i \boldsymbol{v}_i(u) > -(t+1)\lambda_2^t (\lambda_2 - \lambda_k) \sum_{i : \alpha_i \boldsymbol{v}_i(u) < 0} \alpha_i \boldsymbol{v}_i(u). \tag{7}$$

We lower bound the left hand side and upper bound the right hand side. For the lower bound we apply Lemma 2.2 to get that $\sum_{i=2}^k \alpha_i \boldsymbol{v}_i(u) = \boldsymbol{y}(u)$ and Lemma B.1 to get $\boldsymbol{y}(u) \geqslant \frac{k}{\Delta n}$, with high probability. For the upper bound, instead, we rely on Claim B.4 and on the fact that $\alpha_i \boldsymbol{v}_i = \beta_i D^{\frac{1}{2}} \boldsymbol{w}_i$, for every $i \in [n]$. Indeed

$$-\sum_{i : \alpha_i \boldsymbol{v}_i(u) < 0} \alpha_i \boldsymbol{v}_i(u) = -\sum_{i :\ \beta_i \boldsymbol{w}_i(u) < 0} \frac{\beta_i}{\sqrt{\delta(u)}} \boldsymbol{w}_i(u) \overset{(a)}{\leqslant} k\sqrt{\Delta \log n},$$

where in (a) we can apply Claim B.4 since $\|\boldsymbol{w}_i\|_2 = 1$ for every $i \in [k]$ and $\beta_i = \langle D^{\frac{1}{2}}\boldsymbol{x}, \boldsymbol{w}_i\rangle$. By combining lower and upper bounds, we get

$$\begin{aligned}
\frac{1}{2}\lambda_k^t(1-\lambda_2)\sum_{i=2}^{k}\alpha_i\boldsymbol{v}_i(u) &> -(t+1)\lambda_2^t(\lambda_2-\lambda_k)\sum_{i:\alpha_i\boldsymbol{v}_i(u)<0}\alpha_i\boldsymbol{v}_i(u)\\
\frac{1}{2}\lambda_k^t(1-\lambda_2)\frac{k}{\Delta n} &> (t+1)\lambda_2^t(\lambda_2-\lambda_k)k\sqrt{\Delta\log n}\\
\left(\frac{\lambda_2}{\lambda_k}\right)^t(t+1) &< \frac{1}{2}\frac{1-\lambda_2}{\lambda_2-\lambda_k}\frac{1}{\Delta^{\frac{3}{2}}n\sqrt{\log n}}. \qquad (8)
\end{aligned}$$

By hypothesis we have that $\frac{1-\lambda_2}{\lambda_2-\lambda_k} \geqslant \Delta^{\frac{3}{2}}n^{1+c}$ and that $\lambda_k > \frac{1}{2}$. Thus, we can derive an upper bound for $\frac{\lambda_2}{\lambda_k}$, namely

$$\frac{\lambda_2}{\lambda_k} = 1 + \frac{\lambda_2-\lambda_k}{\lambda_k} \leqslant 1 + \frac{1-\lambda_2}{\lambda_k\Delta^{\frac{3}{2}}n^{1+c}} \leqslant 1 + \frac{1}{\Delta^{\frac{3}{2}}n^{1+c}} \leqslant 1 + \frac{1}{n^{\frac{c}{3}}}. \qquad (9)$$

Moreover, by the hypothesis on $\frac{1-\lambda_2}{\lambda_2-\lambda_k}$, we know that

$$\frac{1}{2}\frac{1-\lambda_2}{\lambda_2-\lambda_k}\frac{1}{\Delta^{\frac{3}{2}}n\sqrt{\log n}} \geqslant \frac{1}{2}n^{\frac{c}{2}}. \qquad (10)$$

We apply Equations (9) and (10) to Equation (8) to find a time T_2 such that for every $t \leqslant T_2$ the lemma holds, and get

$$\left(1 + \frac{1}{n^{\frac{c}{3}}}\right)^t(t+1) < \frac{1}{2}n^{\frac{c}{2}}.$$

Let $T_2 = n^{\frac{c}{3}}$. Note that $\left(1+\frac{1}{n^{\frac{c}{3}}}\right)^t \leqslant e$ for every time $t \leqslant T_2$; thus, for every time $t < T_2$, it also holds that $e\,(t+1) < \frac{1}{2}n^{\frac{c}{2}}$. We conclude that, in this first case, there exists a time $T_2 = \Omega(n^{\frac{c}{3}})$ such that, for every $t < T_2$,

$$\boldsymbol{c}^{(t)}(u) - \boldsymbol{c}^{(t+1)}(u) \geqslant \frac{1}{2}\lambda_k^t(1-\lambda_2)\sum_{i=1}^{k-1}\gamma_i\boldsymbol{\chi}_i(u). \qquad (11)$$

Let us now suppose $\boldsymbol{c}^{(t)}(u) - \boldsymbol{c}^{(t+1)}(u) < 0$. As before, if $\boldsymbol{y}(u) > 0$ the thesis directly follows; then let us suppose $\boldsymbol{y}(u) \leqslant 0$. From Claim B.3 we have that

$$\boldsymbol{c}^{(t)}(u) - \boldsymbol{c}^{(t+1)}(u) \leqslant \lambda_k^t(1-\lambda_2)\sum_{i=2}^{k}\alpha_i\boldsymbol{v}_i(u) + (t+1)\lambda_2^t(\lambda_2-\lambda_k)\sum_{i:\alpha_i\boldsymbol{v}_i(u)>0}\alpha_i\boldsymbol{v}_i(u).$$

Similarly to the previous case, in order to prove the lemma we need to show that

$$\frac{1}{2}\lambda_k^t(1-\lambda_2)\sum_{i=2}^{k}\alpha_i\boldsymbol{v}_i(u) \leqslant -(t+1)\lambda_2^t(\lambda_2-\lambda_k)\sum_{i:\alpha_i\boldsymbol{v}_i(u)>0}\alpha_i\boldsymbol{v}_i(u). \qquad (12)$$

Again, we upper bound the left hand side using Lemma 2.2 and Lemma B.1 and getting $\sum_{i=2}^{k}\alpha_i\boldsymbol{v}_i(u) = \sum_{i=1}^{k-1}\gamma_i\boldsymbol{\chi}_i(u) \leqslant -\frac{k}{\Delta n}$, with high probability. As for the right hand side we use Claim B.4 and get that $-\sum_{i:\alpha_i\boldsymbol{v}_i(u)>0}\alpha_i\boldsymbol{v}_i(u) \geqslant -k\sqrt{\Delta\log n}$. By combining the two bounds we get

$$-\frac{1}{2}\lambda_k^t(1-\lambda_2)\frac{k}{\Delta n} < -(t+1)\lambda_2^t(\lambda_2-\lambda_k)k\sqrt{\Delta\log n},$$

which is exactly the same condition of the previous case. Thus, for every time $t < T_2 = \Omega(n^{\frac{3}{2}})$, we have that

$$\boldsymbol{c}^{(t)}(u) - \boldsymbol{c}^{(t+1)}(u) \leqslant \frac{1}{2}\lambda_k^t(1-\lambda_2)\sum_{i=1}^{k-1}\gamma_i\boldsymbol{\chi}_i(u). \tag{13}$$

By combining Eq. (11) and Eq. (13), we conclude that $\left|\boldsymbol{c}^{(t)}(u) - \boldsymbol{c}^{(t+1)}(u)\right| \geqslant \frac{1}{2}\lambda_k^t(1-\lambda_2)\left|\boldsymbol{y}(u)\right|$.

Now we show that $\operatorname{sgn}(\boldsymbol{c}^{(t)}(u)-\boldsymbol{c}^{(t+1)}(u)) = \operatorname{sgn}(\boldsymbol{y}(u))$. In particular, Equations (7) and (12) imply that $-(t+1)\lambda_2^t(\lambda_2-\lambda_k)\sum_{i:\alpha_i\boldsymbol{v}_i(u)<0}\alpha_i\boldsymbol{v}_i(u) \leqslant \frac{1}{2}\lambda_k^t(1-\lambda_2)\left|\boldsymbol{y}(u)\right|$ and that $(t+1)\lambda_2^t(\lambda_2-\lambda_k)\sum_{i:\alpha_i\boldsymbol{v}_i(u)>0}\alpha_i\boldsymbol{v}_i(u) \leqslant \frac{1}{2}\lambda_k^t(1-\lambda_2)\left|\boldsymbol{y}(u)\right|$. Thus, upper and lower bounds for $\boldsymbol{c}^{(t)}(u) - \boldsymbol{c}^{(t+1)}(u)$ in Claim B.3, during for every $t < T_2$, have the same sign of $\boldsymbol{y}$ and consequently $\operatorname{sgn}(\boldsymbol{c}^{(t)}(u) - \boldsymbol{c}^{(t+1)}(u)) = \operatorname{sgn}(\boldsymbol{y}(u))$. □

B.3 Upper bound on the error contribution

Lemma B.3 (Upper bound on the error contribution)**.** *Let* $\boldsymbol{e}^{(t)} := \sum_{i=k+1}^{n}\lambda_i^t\alpha_i\boldsymbol{v}_i$. *For every* $u \in V$, *it holds that*

$$|\boldsymbol{e}^{(t)}(u)| \leqslant \lambda_{k+1}^t\sqrt{\Delta n}.$$

Proof. To bound all components of vector $\boldsymbol{e}^{(t)}$ we use its ℓ^∞ norm, defined for any vector $\boldsymbol{x}$ as $\|\boldsymbol{x}\|_\infty := \sup_i|\boldsymbol{x}(i)|$. In particular

$$\begin{aligned}
\|\boldsymbol{e}^{(t)}\|_\infty^2 \leqslant \|\boldsymbol{e}^{(t)}\|^2 &= \left\|\sum_{i=k+1}^{n}\lambda_i^t\alpha_i\boldsymbol{v}_i\right\|^2 = \left\|\sum_{i=k+1}^{n}\lambda_i^t\beta_i D^{-\frac{1}{2}}\boldsymbol{w}_i\right\|^2 \\
&\overset{(a)}{\leqslant} \left\|D^{-\frac{1}{2}}\right\|^2\left\|\sum_{i=k+1}^{n}\lambda_i^t\beta_i\boldsymbol{w}_i\right\|^2 \overset{(b)}{=} \left\|D^{-\frac{1}{2}}\right\|^2\sum_{i=k+1}^{n}\lambda_i^{2t}\beta_i^2 \\
&\leqslant \left\|D^{-\frac{1}{2}}\right\|^2\lambda_{k+1}^{2t}\sum_{i=k+1}^{n}\beta_i^2 \leqslant \left\|D^{-\frac{1}{2}}\right\|^2\lambda_{k+1}^{2t}\sum_{i=1}^{n}\beta_i^2 \\
&= \left\|D^{-\frac{1}{2}}\right\|^2\lambda_{k+1}^{2t}\left\|D^{\frac{1}{2}}\boldsymbol{x}\right\|^2 \leqslant \left\|D^{-\frac{1}{2}}\right\|^2\lambda_{k+1}^{2t}\left\|D^{\frac{1}{2}}\right\|^2\|\boldsymbol{x}\|^2 \\
&\overset{(c)}{=} \frac{\max_u\delta(u)}{\min_u\delta(u)}\lambda_{k+1}^{2t}\|\boldsymbol{x}\|^2 \leqslant \lambda_{k+1}^{2t}\Delta n,
\end{aligned}$$

where in (a) we use Cauchy-Schwarz inequality (Theorem A.1) and we apply the definition of spectral norm of an operator, i.e., $\|A\| := \sup_{\boldsymbol{x}:\|\boldsymbol{x}=1\|}\|A\boldsymbol{x}\|$; in (b) we use that the $\boldsymbol{w}_i$s are orthonormal; in (c) we use that the spectral norm of a diagonal matrix is equal to its maximum value. Thus, for every $u \in V$ it holds that $|\boldsymbol{e}^{(t)}(u)| \leqslant \sqrt{\|\boldsymbol{e}^{(t)}\|_\infty^2} \leqslant \lambda_{k+1}^t\sqrt{\Delta n}$. □