A spectral theorem on the cluster structure of real-world graphs

Sabyasachi Basu
Department of Computer Science and Engineering, University of California, Santa Cruz

Suman Kalyan Bera
Katana Graph

C. Seshadhri
Department of Computer Science and Engineering, University of California, Santa Cruz

Abstract

Partitioning a graph into clusters of vertices is a fundamental problem in computer science and applied mathematics. Arguably, the most important tool for graph partitioning is the Fielder vector or discrete Cheeger inequality. This result relates the eigenvalues of the normalized adjacency matrix to the low conductance cuts of the graph. However, the Cheeger inequality has little relevance on an important contemporary graph partitioning problem, that of community detection in massive real-world graphs. There are numerous, small, dense clusters in real-world graphs, while Cheeger inequalities focus on partitioning a graph into a few, large clusters. Inspired by the structure of real-world graphs, we define the *spectral transitivity*, a ratio of powers of eigenvalues of the normalized adjacency matrix $\mathcal{A}$. We discover that constant spectral transitivity implies that a constant fraction of $\mathcal{A}$ is contained in nearly uniform submatrices. Our result is a new spectral theorem that relates the eigenvalues of $\mathcal{A}$ to a cluster structure in $\mathcal{A}$. The latter structure mimics the observed cluster structure of real-world graphs.

2012 ACM Subject Classification Mathematics of computing → Graph algorithms

Keywords and phrases Graph partitioning, Spectral graph theory, Social networks

Digital Object Identifier 10.4230/LIPIcs...1

1 Introduction

Graph partitioning or clustering is a fundamental problem in theoretical computer science. It has a rich history in the study of algorithms, applied mathematics, and computer science. One of the central tools in graph partitioning is the discrete Cheeger inequality, which goes back to seminal work of Fiedler, and Alon and Milman [9, 1]. This inequality relates the eigenvalues of the graph Laplacian to the graph conductance, showing a connection between the spectrum and graph structure. Consider an undirected graph $G = (V, E)$, where d_v denotes the degree of vertex v. The *normalized adjacency matrix*, denoted $\mathcal{A}$, is defined as follows: $\mathcal{A}_{uv} = 1/\sqrt{d_u d_v}$ if $(u, v) \in E$, and zero otherwise. The eigenvalues of this matrix are denoted $1 = \lambda_1 \geq \lambda_2 \geq \lambda_3 \ldots \lambda_n \geq -1$.

We recall the definition of graph *conductance*. For any subset of vertices $S \subseteq V$, let $\mathrm{Vol}(S) := \sum_{s \in S} d_s$. The conductance of set S is $\Phi(S) := E(S, \overline{S})/\min(\mathrm{Vol}(S), \mathrm{Vol}(\overline{S}))$. The conductance of the graph G, Φ_G, is defined as $\min_{S \subseteq V} \Phi(S)$. The classic Cheeger inequality relates the spectral gap, $1 - \lambda_2$, to the graph conductance.

▶ **Theorem 1.1.** *(Cheeger inequality [6, 18])* $4\sqrt{1 - \lambda_2} \geq \Phi_G \geq (1 - \lambda_2)/4$

This theorem is the foundation of spectral graph theory. The proof also yields an efficient algorithm that finds a low conductance cut.

One of the most important contemporary applications of graph clustering is *community detection* in real-world sparse graphs [25, 24, 23, 10, 11]. Despite the wide applicability of the

Cheeger inequality in general, it is surprisingly irrelevant for network science applications. Firstly, conductance pertains to breaking the graph into two parts. There are higher order Cheeger inequalities that deal with k parts, but these are only applicable for $k = O(\log n)$ [20]. Real-world graphs have an extremely large number of small clusters, each of which is internally dense [21, 29]. Variants of the Cheeger inequalities cannot deal with large k (say, n^δ) and do not give edge density guarantees about the interior of clusters. We note that there are local partitioning theorems inspired by the proof of the Cheeger inequality that find small clusters or give some guarantees on internal structure [37, 20, 26, 27]. But these results do not connect the graph spectrum to graph structure.

Diffusion/PageRank based methods on real-world graphs find a large number of small sets with conductances around 0.1 or so [21, 13]. For real-world graphs, the connection between spectral gap and conductance does not seem to be the central theme. In fact, the commonly observed small world property implies a fairly large spectral gap [19]. Most real-world networks have a significant fraction of long-range edges or weak ties, that are *not* part of any community [22, 14, 19]. These edges essentially make the graph be an expander, in which case the Cheeger inequality has little to say. The spectral gap is sensitive to noise, so adding (say) a sparse Erdős-Rényi graph (or a set of random edges) on top of an existing graph could dramatically change the spectral graph and conductances. But that is exactly what is used for certain models for social networks [29].

Our main motivation is:

Can we relate the graph spectrum to the cluster properties of real-world graphs?

1.1 Main result

We take inspiration from a central property of real-world graphs, the abundance of triangles [36, 29]. This abundance is widely seen across graphs that come by disparate domains. Recent work in network science and data mining have used the triangles to effectively cluster graphs. There is much evidence that the triangle structure aids finding communities in graphs [28, 33, 3, 34].

In network science, the triangle count is often expressed in terms of the *transitivity* or global clustering coefficient [8, 35]. We define the *spectral transitivity* of the graph G.

► **Definition 1.2.** *The spectral transitivity of G, denoted $\tau(G)$, is defined as follows*[1]*. (Recall that the λ_is are the eigenvalues of the normalized adjacency matrix.)*

$$\tau(G) = \frac{\sum_{i \leq n} \lambda_i^3}{\sum_{i \leq n} \lambda_i^2}. \tag{1}$$

Standard arguments show that the spectral transitivity is a degree weighted transitivity. The numerator is a weighted sum over all triangles, while the denominator (squared Frobenius norm) is a weighted sum over edges (Lemma 3.5).

Observe that since $\lambda_i \leq 1$, $\tau \leq 1$. When τ reaches its maximum value of $1 - 1/(n-1)$, one can show that G is a clique (Lemma 3.6). We formalize the notion of "clique-like" submatrices through the concept of uniformity. For a symmetric matrix M and a subset S of its columns/rows, we use $M|_S$ to denote the square submatrix restricted to S (on both columns and rows).

[1] If G (or the normalized adjacency matrix $\mathcal{A}$) are obvious from context, we simply refer to τ instead of $\tau(G)$.

▶ **Definition 1.3.** *Let $\alpha \in (0,1]$. Let $\mathcal{A}$ be the normalized adjacency matrix of a graph G. For any subset of vertices S, $|\mathcal{A}|_S$ is called α-strongly uniform if at least an α-fraction of non diagonal entries have values in the range $[\alpha/(|S|-1), 1/\alpha(|S|-1)]$.*

For $s \in S$, let $N(s,S)$ denote the neighborhood of s in S (we define edges by non-zero entries). An α-uniform matrix is strongly α-uniform *if for at least an α-fraction of $s \in S$, $\mathcal{A}|_{N(s,S)}$ is also α-uniform.*

Observe that the normalized adjacency matrix of a clique is (strongly) 1-uniform. But submatrices of this matrix are not. Roughly speaking, a constant uniform submatrix corresponds to a dense subgraph of (say) size k where the *total* degrees of vertices is $\Theta(k)$. Strong uniformity is closely related to *clustering coefficients*, which is the edge density of neighborhoods. It is well-known that real-world graphs have high clustering coefficients [36, 29]. A strongly uniform submatrix essentially exhibits high clustering coefficients.

Our main theorem states that any graph with constant spectral transitivity can be decomposed into constant uniform blocks. We use $\|M\|_2$ to denote the Frobenius norm of matrix M.

▶ **Theorem 1.4** (Spectral Theorem). *There exist absolute constants $\delta > 0$ and $c > 0$ such that the following holds. Let $\mathcal{A}$ be the normalized adjacency matrix of a graph with spectral transitivity τ.*

There exists a collection of disjoint sets of vertices $X_1, X_2, \ldots, X_k$ satisfying the following conditions:

1. *(Cluster structure) For all $i \leq k$, $\mathcal{A}|_{X_i}$ is strongly $\delta\tau^c$-uniform.*
2. *(Coverage) $\sum_{i \leq k} \|\mathcal{A}|_{X_i}\|_2^2 \geq \delta\tau^c \|\mathcal{A}\|_2^2$.*

We call this output the *spectral triadic decomposition*. Our proof also yields an efficient algorithm that computes the decomposition, whose running time is dominated by a triangle enumeration. Details in are given in Theorem 6.1 and §6.

1.2 Significance of Theorem 1.4

One can think of Theorem 1.4 as a type of Cheeger inequality that is relevant to the structure of real-world social networks. We explain how it captures many of the salient properties of clusters in real-world networks. In this discussion, we will assume that τ is a constant.

The spectral transitivity: We find it remarkable that a bound on a single spectral quantity, τ, implies such a rich decomposition. The spectral transitivity τ captures a key property of real-world graphs, the abundance of triangles. While there is a rich body of empirical work on using triangles to cluster graphs, there is no theory explaining *why* triangles are so useful. Theorem 1.4 gives a spectral-theoretic explanation.

The spectral transitivity is a weighted version of the transitivity, which is typically around 0.1 for real-world graphs[2]. We also note that the final algorithm that computes the decomposition focuses on triangle cuts, which is a popular empirical technique for finding clusters in social networks [3, 34].

The strong uniformity of clusters: Each cluster X_i of the spectral triadic decomposition is (constant) strongly uniform. While there is no one definition of a "community" in real-world graphs, the definition of strong uniformity captures many basic concepts. Most

[2] Our experiments on these real-world graphs yield similar values for the spectral transitivity.

importantly, X_i is internally dense in edges. Let $|X_i| = k$. Then $\Omega(k^2)$ entries in X_i are $\Omega(1/k)$, which (by averaging) implies that a constant fraction of X_i involves vertices of degree $\Theta(k)$. Thus, a constant fraction of X_i vertices have a constant fraction of their neighbors in X_i. Moreover, the submatrix of every neighborhood in X_i is also uniform. This is quite consistent with the typical notion of a social network community.

Crucially, Theorem 1.4 gives a condition on the *internal* structure of the decomposition. This addresses a key weakness of the Cheeger inequality.

The coverage condition: It is natural to measure the "mass" of a matrix by the squared Frobenius norm. The clusters of spectral triadic decomposition of Theorem 1.4 capture a constant fraction of this squared norm. This is consistent with the fact that a constant fraction of the edges in a real-world graph are *not* community edges [22, 14, 19, 29]. Any decomposition into communities would avoid these "long-range" edges, excluding a constant fraction of the matrix mass.

Robustness to noise: Taking the above point further, the non-community edges are often modeled as stochastic (or noisy). The underlying cluster structure of a real-world graph is robust to such perturbations. Adding (say) an Erdős-Rényi graph with $\Theta(n)$ edges can only affect the spectral transitivity by a constant factor (by changing the Frobenius norm). Theorem 1.4 would only be affected by constant factors. Note that the spectral gap, on the other hand, can dramatically increase by such noise.

Spectral graph theory inspired by real-world graphs: We consider Theorem 1.4 as opening up a new direction in spectral graph theory. At a mathematical level, Theorem 1.4 is like a Cheeger inequality, where a spectral condition implies a graph theoretic property. But all aspects of Theorem 1.4 (the notion of spectral transitivity and the properties of the decomposition) are inspired by the observed properties of real-world graphs.

2 Related Work

Spectral graph theory is a deep field of study with much advancement over the past two decades. We refer the readers to the classic textbook by Chung [7], and the tutorial [31] and lecture notes [30] by Spielman.

The cluster structure of real-world networks has attracted attention from the early days of network science [12, 23]. Fortunato's (somewhat dated) survey on community detection has details of the key results [10]. There is no definitive model for social networks, but it is generally accepted that they have many dense clusters with sparse connections between them [5, 21, 29]. The study of triangles and neighborhood density goes back to the early days of social science theory [16, 17, 4, 8]. Early network science papers popularized the notion of clustering coefficients and transitivity as useful measures [36]. The use of triangles to find such clusters is a more recent development in network science. A number of contemporary results explicit use triangle information for algorithmic purposes [28, 33, 3, 34]. Our main theorem is inspired by these applications.

While the Cheeger inequality by itself is not useful for real-world graph clustering, local versions of spectral clustering are extremely useful [32, 2]. We stress that these results do not relate the graph spectrum to the partitions. But the algorithm is inspired by the proof of the Cheeger inequality. Many results on the cluster structure of real-world graphs [21, 13] use the Personalized PageRank method [2]. Some local partitioning methods yield bounds on the internal structure of clusters [20, 26, 27].

Most relevant to our work is the result of Gupta, Roughgarden, and Seshadhri [15]. They prove a decomposition theorem for triangle-rich graphs, as measured by graph transitivity.

Their main result shows that a triangle-dense graph can be clustered into dense clusters. The results of [15] do not have any spectral connection, nor do they provide the kind of uniformity or coverage bounds of Theorem 1.4. Our main insight is in generalizations of their proof technique, which leads to connections with graph spectrum. We adapt the [15] proof to deal with normalized adjacency matrix, which adds many complications because of the non-uniformity of entries.

3 Preliminaries

We use V, E, T to denote the sets of vertices, edges, and triangles of G, respectively. For any subgraph H of G, we use V_H, E_H, T_H to denote the corresponding sets within H. For any edge e, let $T_H(e)$ denote the set of triangles in H containing e.

For any vertex v, let d_v denote the degree of v (in G).

We first define the notion of *weights* for edges and triangles. We will think of edges and triangles as unordered sets of vertices.

▶ **Definition 3.1.** *For any edge $e = (u, v)$, define the weight* $\mathrm{wt}(e)$ *to be* $\frac{1}{d_u d_v}$*. For any triangle $t = (u, v, w)$, define the weight* $\mathrm{wt}(t)$ *to be* $\frac{1}{d_u d_v d_w}$*.*

For any set S consisting solely of edges or triangles, define $\mathrm{wt}(S) = \sum_{s \in S} \mathrm{wt}(s)$*.*

We state some basic facts that relate the sum of weights to sum of eigenvalue powers. Let $S \subset V$ be any subset of vertices, and let $\mathcal{A}|_S$ denote the submatrix of $\mathcal{A}$ restricted to S. We use $\lambda_i(S)$ to denote the ith largest eigenvalue of the symmetric submatrix $\mathcal{A}|_S$. Abusing notation, we use E_S and T_S to denote the edges and triangles contained in the graph induced on S.

▷ Claim 3.2. $\sum_{i \le |S|} \lambda^2(S)_i = 2 \sum_{e \in E(S)} \mathrm{wt}(e)$

Proof. By the properties of the Frobenius norm of matrices, $\sum_{i \le |S|} \lambda_i^2 = \sum_{s,t \in S} \mathcal{A}_{st}^2$. Note that $\mathcal{A}_{st} = A_{st}/\sqrt{d_s d_t}$. Hence, $\sum_{s,t} \mathcal{A}_{s,t}^2 = 2 \sum_{e=(u,v) \in E(S)} 1/d_u d_v$. (We get a 2-factor because each edge (u, v) appears twice in the adjacency matrix.) ◀

▷ Claim 3.3. $\sum_{i \le |S|} \lambda^3(S)_i = 6 \sum_{t \in T(S)} \mathrm{wt}(t)$.

Proof. Note that $\sum_{i \le |S|} \lambda^3(S)_i$ is the trace of $(\mathcal{A}|_S)^3$. The diagonal entry $(\mathcal{A}|_S)_{ii}^3$ is precisely $\sum_{s \in S} \sum_{s' \in S} \mathcal{A}_{is} \mathcal{A}_{ss'} \mathcal{A}_{s'i}$. Note that $\mathcal{A}_{is} \mathcal{A}_{ss'} \mathcal{A}_{s'i}$ is non-zero iff (i, s, s') form a triangle. In that case, $\mathcal{A}_{is} \mathcal{A}_{ss'} \mathcal{A}_{s'i} = 1/\sqrt{d_i d_s} \cdot 1/\sqrt{d_s d_{s'}} \cdot 1/\sqrt{d_{s'} d_i} = \mathrm{wt}((i, s, s'))$. We conclude that $(\mathcal{A}|_S)_{ii}^3$ is $2 \sum_{t \in T(S), t \ni i} \mathrm{wt}(t)$. (There is a 2 factor because every triangle is counted twice.)

Thus, $\sum_{i \le n} \lambda^3(S)_i = \sum_i 2 \sum_{t \in T, t \ni i} \mathrm{wt}(t) = 2 \sum_{t \in T} \sum_{i \in t} \mathrm{wt}(t) = 6 \sum_{t \in T} \mathrm{wt}(t)$. (The final 3 factor appears because a triangle contains exactly 3 vertices.) ◀

▷ Claim 3.4. $\sum_{t \in T(S)} \mathrm{wt}(t) \le \|\mathcal{A}|_S\|_2^2/6$.

Proof. By Claim 3.3 $\sum_{t \in T(S)} \mathrm{wt}(t) = \sum_{i \le |S|} \lambda^3(S)_i/6$. The maximum eigenvalue of $\mathcal{A}$ is 1, and since $\mathcal{A}|_S$ is a submatrix, $\lambda(S)_1 \le 1$ (Cauchy's interlacing theorem). Thus, $\sum_{i \le |S|} \lambda^3(S)_i \le \sum_{i \in |S|} \lambda^2(S)_i = \|\mathcal{A}|_S\|_2^2$. ◀

As a direct consequence of the previous claims applied on $\mathcal{A}$, we get the following characterization of the spectral triadic content in terms of the weights.

▶ **Lemma 3.5.** $\tau = \frac{3 \sum_{t \in T} \mathrm{wt}(t)}{\sum_{e \in E} \mathrm{wt}(e)}$.

While the following bound is not necessary for our main result, it is instructive to see the largest possible value of the spectral transitivity.

▶ **Lemma 3.6.** *Consider normalized adjacency matrices $\mathcal{A}$ with n vertices. The maximum value of $\tau(\mathcal{A})$ is $1 - 1/(n-1)$. This value is attained for the unique strongly 1-uniform matrix, the normalized adjacency matrix of the n-clique.*

Proof. First, consider the normalized adjacency matrix $\mathcal{A}$ of the n-clique. All off-diagonal entries are precisely $1/(n-1)$ and $\mathcal{A}$ can be expressed as $(n-1)^{-1}(\mathbf{1}\mathbf{1}^T - I)$. The matrix $\mathcal{A}$ is 1-regular. The largest eigenvalue is 1 and all the remaining eigenvalues are $-1/(n-1)$. Hence, $\sum_i \lambda_i^3 = 1 - (n-1)/(n-1)^3 = 1 - 1/(n-1)^2$. The sum of squares of eigenvalue is $\sum_i \lambda_i^2 = 1 + (n-1)/(n-1)^2 = 1 + 1/(n-1)$. Dividing,

$$\frac{\sum_{i \le n} \lambda_i^3}{\sum_{i \le n} \lambda_i^2} = 1 - 1/(n-1).$$

Since the matrix has zero diagonal, the trace $\sum_i \lambda_i$ is zero. We will now prove the following claim.

▷ Claim 3.7. Consider any sequence of numbers $1 = \lambda_1 \ge \lambda_2 \ldots \ge \lambda_n$ such that $\forall i, |\lambda_i| \le 1$ and $\sum_i \lambda_i = 0$. If $\sum_i \lambda_i^3 \ge (1 - 1/(n-1)) \sum_i \lambda_i^2$, then $\forall i > 1, \lambda_i = -1/(n-1)$.

Proof. Let us begin with some basic manipulations.

$$\sum_i \lambda_i^3 \ge [1 - 1/(n-1)] \sum_i \lambda_i^2 \tag{2}$$

$$\implies 1 + \sum_{i>1} \lambda_i^3 \ge [1 - 1/(n-1)] \cdot (1 + \sum_{i>1} \lambda_i^2)$$

$$\implies \sum_{i>1} \lambda_i^3 \ge [1 - 1/(n-1)] \sum_{i>1} \lambda_i^2 - 1/(n-1). \tag{3}$$

For $i > 1$, define $\delta_i := \lambda_i + 1/(n-1)$. Note that $\sum_{i>1} \lambda_i = -1$, so $\sum_{i>1} \delta_i = 0$. Moreover, $\forall i > 1$, $\delta_i \le 1 + 1/(n-1)$. We plug in $\lambda_i = \delta_i - 1/(n-1)$ in (3).

$$\sum_{i>1} \Big[\delta_i - 1/(n-1)\Big]^3 \ge [1 - 1/(n-1)] \sum_{i>1} \Big[\delta_i - 1/(n-1)\Big]^2 - 1/(n-1)$$

$$\implies \sum_{i>1} \Big[\delta_i^3 - 3\delta_i^2/(n-1) + 3\delta_i/(n-1)^2 - 1/(n-1)^3\Big]$$

$$\ge [1 - 1/(n-1)] \sum_{i>1} \Big[\delta_i^2 - 2\delta_i/(n-1) + 1/(n-1)^2\Big] - 1/(n-1).$$

Recall that $\sum_{i>1} \delta_i = 0$. Hence, we can simplify the above inequality.

$$\sum_{i>1} \delta_i^3 - (3/(n-1)) \sum_{i>1} \delta_i^2 - 1/(n-1)^2$$

$$\ge [1 - 1/(n-1)] \sum_{i>1} \delta_i^2 + 1/(n-1) - 1/(n-1)^2 - 1/(n-1)$$

$$\implies \sum_{i>1} \delta_i^3 \ge [1 + 2/(n-1)] \sum_{i>1} \delta_i^2. \quad \text{(Canceling terms and rearranging)}$$

Since $\delta_i \le (1 + 1/(n-1))$, we get that $\sum_{i>1} \delta_i^3 \le [1 + 1/(n-1)] \sum_{i>1} \delta_i^2$. Combining with the above inequality, we deduce that $[1 + 2/(n-1)] \sum_{i>1} \delta_i^2 \le [1 + 1/(n-1)] \sum_{i>1} \delta_i^2$. This can only happen if $\sum_{i>1} \delta_i^2$ is zero, implying all δ_i values are zero. Hence, for all $i > 1$, $\lambda_i = -1/(n-1)$. ◀

With this claim, we conclude that any matrix $\mathcal{A}$ maximizing the ratio of cubes and squares of eigenvalues has a fixed spectrum. It remains to prove that a unique normalized adjacency matrix has this spectrum. We use the rotational invariance of the Frobenius norm: sum of squares of entries of $\mathcal{A}$ is the same as the sum of squares of eigenvalues. Thus,

$$\sum_{(u,v)\in E} \frac{2}{d_u d_v} = 1 + \frac{1}{n-1} = \frac{n}{n-1}. \tag{4}$$

Observe that $\frac{2}{d_u d_v} \geq 1/(d_u(n-1)) + 1/(d_v(n-1))$, since all degrees are at most $n-1$. Summing this inequality over all edges,

$$\sum_{(u,v)\in E} \frac{2}{d_u d_v} \geq \sum_{v\in V} \sum_{u\in N(v)} \frac{1}{d_v(n-1)} = \sum_{v\in V} \frac{d_v}{d_v(n-1)} = \frac{n}{n-1}. \tag{5}$$

Hence, for (4) to hold, for all edges (u,v), we must have the equality $\frac{2}{d_u d_v} = 1/(d_u(n-1)) + 1/(d_v(n-1))$. That implies that for all edge (u,v), $d_u = d_v = n-1$. So all vertices have degree $(n-1)$, and the graph is an n-clique. ◀

We will need the following "reverse Markov inequality" for some intermediate proofs.

▶ **Lemma 3.8.** *Consider a random variable Z taking values in $[0,b]$. If $\mathbf{E}[Z] \geq \sigma b$, then $\Pr[Z \geq \sigma b/2] \geq \sigma/2$.*

Proof. In the following calculations, we will upper bound the conditional expectation by the maximum value (under that condition).

$$\begin{aligned}
\sigma b \leq \mathbf{E}[Z] &= \Pr[Z \geq \sigma b/2] \cdot \mathbf{E}[Z|Z \geq \sigma b/2] + \Pr[Z & (6)\\
&\leq \sigma b/2] \cdot \mathbf{E}[Z|Z \leq \sigma b/2] & (7)\\
&\leq \Pr[Z \geq \sigma b/2] \cdot b + \sigma b/2 & (8)
\end{aligned}$$

We rearrange to complete the proof. ◀

4 Cleaned graphs and extraction

For convenience, we set $\varepsilon = \tau/6$.

▶ **Definition 4.1.** *A connected subgraph H is called* clean *if $\forall e \in E(H)$, $\mathrm{wt}(T_H(e)) \geq \varepsilon \mathrm{wt}(e)$.*

Algorithm 1 `Extract`(H)

1: Pick $v \in V(H)$ that minimizes d_v.
2: Construct the set $L := \{u|(u,v) \in E(H), d_u \leq 2\varepsilon^{-1} d_v\}$ (L is the set of low degree neighbors of v in H.)
3: For every vertex $w \in V(H)$, define ρ_w to be the total weight of triangles of the form (w,u,u') where $u,u' \in L$.
4: Sort the vertices in decreasing order of ρ_w, and construct the "sweep cut" C to be the smallest set satisfying $\sum_{w\in C} \rho_w \geq (1/2) \sum_{w\in V(H)} \rho_w$.
5: Output $X := \{v\} \cup L \cup C$.

The main theorem of this section follows.

▶ **Theorem 4.2.** *Suppose the subgraph H is connected and clean. Let X denote the output of the procedure* `Extract`(H). *Then*

$$\sum_{t \in T(H), t \subseteq X} \mathrm{wt}(t) \geq (\varepsilon^8/2000) \sum_{t \in T(H), t \cap X \neq \emptyset} \mathrm{wt}(t)$$

(The triangle weight contained inside X is a constant fraction of the triangle weight incident to X.)

Moreover, $\mathcal{A}|_X$ is strongly $\delta\varepsilon^{12}$-uniform.

We will need numerous intermediate claims to prove this theorem. We use v, L, and C as defined in `Extract`(H). We use N to denote the neighborhood of v in H. Note that $L \subseteq N$.

For any vertex $u \in N$, we define the set of partners $P(u)$ to be $\{w : (u, v, w) \in T_H\}$.

The following lemma is an important tool in our analysis.

▶ **Lemma 4.3.** *For any $u \in N$, $\sum_{w \in P(u) \cap L} d_w^{-1} \geq \varepsilon/2$.*

Proof. Let $e = (u, v)$. Since H is clean, $\mathrm{wt}(T_H(e)) \geq \varepsilon \mathrm{wt}(e)$. Expanding out the definition of weights,

$$\sum_{w:(u,v,w) \in T_H} \frac{1}{d_u d_v d_w} \geq \frac{\varepsilon}{d_u d_v} \quad \Longrightarrow \quad \sum_{w \in P(u)} d_w^{-1} \geq \varepsilon. \tag{9}$$

Note that L (as constructed in `Extract`(H)) is the subset of N consisting of vertices with degree at most $2\varepsilon^{-1} d_v$. For $w \in N \setminus L$, we have the lower bound $d_w \geq 2\varepsilon^{-1} d_v$. Hence,

$$\sum_{w \in N \setminus L} d_w^{-1} \leq |N \setminus L| (\varepsilon/2) d_v^{-1} \leq d_v \times (\varepsilon/2) d_v^{-1} = \varepsilon/2. \tag{10}$$

In the calculation below, we split the sum of (9) into the contribution from L and from outside L. We apply (10) to bound the latter contribution.

$$\varepsilon \leq \sum_{w \in P(u)} d_w^{-1} \leq \sum_{w \in P(u) \cap L} d_w^{-1} + \sum_{w \in N \setminus L} d_w^{-1} \leq \sum_{w \in P(u) \cap L} d_w^{-1} + \varepsilon/2. \tag{11}$$

◀

▷ Claim 4.4. $|L| \geq \varepsilon d_v/2$

Proof. Since H is connected, there must exist some edge $e = (u, v) \in E(H)$. By Lemma 4.3, $\sum_{w \in P(u) \cap L} d_w^{-1} \geq \varepsilon/2$. Hence, $\sum_{w \in L} d_w^{-1} \geq \varepsilon/2$. Since v is the vertex in $V(H)$ minimizing d_v, for any $w \in V(H)$, $d_w \geq d_v$. Thus,

$$\varepsilon/2 \leq \sum_{w \in L} d_w^{-1} \leq \sum_{w \in L} d_v^{-1} = |L| d_v^{-1}. \tag{12}$$

◀

▷ Claim 4.5. $\sum_{e \in E(H), e \subseteq L} \mathrm{wt}(e) \geq \varepsilon^2/8$.

Proof. By Lemma 4.3, $\forall w \in L$, $\sum_{w' \in P(w) \cap L} d_{w'}^{-1} \geq \varepsilon/2$. We multiply both sides by d_w^{-1} and sum over all $w \in L$.

$$\sum_{w \in L} \sum_{w' \in P(w) \cap L} (d_w d_{w'})^{-1} \geq (\varepsilon/2) \sum_{w' \in L} d_{w'}^{-1}. \tag{13}$$

By Lemma 4.3, $\sum_{w' \in L} d_{w'}^{-1} \geq \varepsilon/2$. Note that $w' \in P(w)$ only if $(w, w') \in E(H)$. Hence, $\sum_{w \in L} \sum_{w' \in L, (w,w') \in E(H)} \mathrm{wt}((w, w')) \geq \varepsilon^2/4$. Note that the summation counts all edges twice, so we divide by 2 to complete the proof. ◀

We now come to the central calculations of the main proof. Recall, from the description of `Extract`, that ρ_w is the total triangle weight of the triangles (w, u, u'), where $u, u' \in L$. We will prove that $\sum_w \rho_w$ is large; moreover, there are a few entries that dominate the sum. The latter bound is crucial to arguing that the sweep set C is not too large.

▷ Claim 4.6. $\sum_{w \in V(H)} \rho_w \geq \varepsilon^3/8$.

Proof. Note that $\sum_{w \in V(H)} \rho_w$ is equal to $\sum_{e \in E(H), e \subset L} \mathrm{wt}(T_H(e))$. Both these expressions give the total weight of all triangles in H that involve two vertices in L. Since H is clean, for all edges $e \in E(H)$, $\mathrm{wt}(T_H(e)) \geq \varepsilon \mathrm{wt}(e)$. Hence, $\sum_{e \in E(H), e \subset L} \mathrm{wt}(T_H(e)) \geq \varepsilon \sum_{e \in E(H), e \subset L} \mathrm{wt}(e)$. Applying Claim 4.5, we can lower bound the latter by $\varepsilon^3/8$. ◀

We now show that a few ρ_w values dominate the sum, using a somewhat roundabout argument. We upper bound the sum of square roots.

▷ Claim 4.7. $\sum_{w \in V(H)} \sqrt{\rho_w} \leq 2\varepsilon^{-1}\sqrt{d_v}$

Proof. Let c_w be the number of vertices in L that are neighbors (in H) of w. Note that for any triangle (u, u', w) where $u, u' \in L$, both u and u' are common neighbors of w and v. The number of triangles (u, u', w) where $u, u' \in L$ is at most c_w^2. The weight of any triangle in H is at most d_v^{-3}, since d_v is the lowest degree (in G) of all vertices in H. As a result, we can upper bound $\rho_w \leq d_v^{-3} c_w^2$.

Taking square roots and summing over all vertices,

$$\sum_{w \in V(H)} \sqrt{\rho_w} \leq d_v^{-3/2} \sum_{w \in V(H)} c_w \tag{14}$$

Note that $\sum_{w \in V(H)} c_w$ is exactly the sum over $u \in L$ of the degrees of u in the subgraph H. (Every edge incident to $u \in L$ gives a unit contribution to the sum $\sum_{w \in V(H)} c_w$.) By definition, every vertex in L has degree in H at most $2\varepsilon^{-1} d_v$. The size of L is at most d_v.

Hence, $\sum_{w \in V(H)} c_w \leq 2\varepsilon^{-1} d_v^2$. Plugging into (14), we deduce that $\sum_{w \in V(H)} \sqrt{\rho_w} \leq 2\varepsilon^{-1}\sqrt{d_v}$. ◀

We now prove that the sweep cut C is small, which is critical to proving Theorem 4.2.

▷ Claim 4.8. $|C| \leq 144\varepsilon^{-5} d_v$.

Proof. For convenience, let us reindex vertices so that $\rho_1 \geq \rho_2 \geq \rho_3 \ldots$. Let $r \leq n$ be an arbitrary index. Because we index in non-increasing order, note that $\sum_{j \leq n} \rho_j \geq r\rho_r$. Furthermore, $\forall j > r$, $\rho_j \leq \rho_r$.

$$\sum_{j>r} \rho_j \leq \sqrt{\rho_r} \sum_{j>r} \sqrt{\rho_j} \leq \sqrt{\frac{\sum_{j \leq n} \rho_j}{r}} \sum_{j \leq n} \sqrt{\rho_j} = \Big[\frac{\sum_{j \leq n} \sqrt{\rho_j}}{\sqrt{r} \cdot \sqrt{\sum_{j \leq n} \rho_j}}\Big] \sum_{j \leq n} \rho_j \tag{15}$$

Observe that Claim 4.7 gives an upper bound on the numerator, while Claim 4.6 gives a lower bound on (a term in) the denominator. Plugging those bounds in (15),

$$\sum_{j>r} \rho_j \leq \frac{2\varepsilon^{-1}\sqrt{d_v}}{\sqrt{r} \cdot \varepsilon^{3/2}/\sqrt{8}} \sum_{j \leq n} \rho_j \leq \frac{1}{\sqrt{r}} \cdot \frac{6\sqrt{d_v}}{\varepsilon^{5/2}} \cdot \sum_{j \leq n} \rho_j. \tag{16}$$

Suppose $r > 144\varepsilon^{-5} d_v$. Then $\sum_{j>r} \rho_j < (1/2) \sum_{j \leq n} \rho_j$. The sweep cut C is constructed with the smallest value of r such that $\sum_{j>r} \rho_j < (1/2) \sum_{j \leq n} \rho_j$. Hence, $|C| \leq 144\varepsilon^{-5} d_v$. ◀

An additional technical claim we need bounds the triangle weight incident to a single vertex.

▷ Claim 4.9. For all vertices $u \in V(H)$, $\mathrm{wt}(T_H(u)) \leq (2d_v)^{-1}$.

Proof. Consider edge $(u,w) \in E(H)$. We will prove that $\mathrm{wt}(T_H((u,w))) \leq d_u^{-1} d_v^{-1}$. Recall that d_v is the smallest degree among vertices in H. Furthermore, $|T_H((u,w))| \leq d_w$, since the third vertex in a triangle containing (u,w) is a neighbor of w.

$$\mathrm{wt}(T_H((u,v))) = \sum_{z:(z,u,w)\in T(H)} \frac{1}{d_u d_w d_z} \leq \frac{1}{d_u d_v} \sum_{z:(z,u,w)\in T(H)} \frac{1}{d_w} \leq \frac{1}{d_u d_v} \times \frac{d_w}{d_w} = \frac{1}{d_u d_v}$$

We now bound $\mathrm{wt}(T_H(u))$ by summing over all neighbors of u in H.

$$\begin{aligned}
\mathrm{wt}(T_H(u)) &= (1/2) \sum_{w:(u,w)\in E(H)} \mathrm{wt}(T_H((u,w))) \\
&\leq (1/2) \sum_{w:(u,w)\in E(H)} \frac{1}{d_u d_v} = \frac{1}{2d_v} \sum_{w:(u,w)\in E(H)} \frac{1}{d_u} \\
&\leq \frac{1}{2d_v} \times \frac{d_u}{d_u} = \frac{1}{2d_v}.
\end{aligned}$$

◀

4.1 The proof of Theorem 4.2

Proof. (of Theorem 4.2) By construction of X as $\{v\} \cup L \cup C$, all the triangles of the form (w,u,u'), where $w \in C$ and $u,u' \in L$, are contained in X. The total weight of such triangles is at least $\sum_{v\leq n} \rho_v/2$, by the construction of C. By Claim 4.6, $\sum_{v\leq n} \rho_v/2 \geq \varepsilon^3/16$.

Let us now bound that total triangle weight incident to X in H. Observe that $|X| = 1 + |L| + |C|$ which is at most $1 + d_v + \varepsilon^{-5}144d_v$, by Claim 4.8. We can further bound $|X| \leq \varepsilon^{-5}146d_v$. By Claim 4.9, the total triangle weight incident to a vertex is at most $(2d_v)^{-1}$. Hence, the total triangle weight incident to all of X is at most $73\varepsilon^{-5}$.

Thus, the triangle weight contained in X is at least $\frac{\varepsilon^3/16}{73\varepsilon^{-5}}$ times the triangle weight incident to X. The ratio is at least $\varepsilon^8/2000$, completing the proof of the first statement.

Proof of uniformity of $\mathcal{A}|_X$: We first prove a lower bound on the uniformity of $\mathcal{A}|_X$. For convenience, let B denote the set $\{e|e \in E(H), e \subseteq L$. By Claim 4.5, $\sum_{e\in B} \mathrm{wt}(e) \geq \varepsilon^2/8$. There are at most $\binom{d_v}{2} \leq d_v^2/2$ edges in B. For every edge e, $\mathrm{wt}(e) \leq 1/d_v^2$. Let k denote the number of edges in B whose weight is at least $\varepsilon^2/16$.

$$\begin{aligned}
\frac{\varepsilon^2}{8} &\leq \sum_{\substack{e\in B \\ \mathrm{wt}(e)\leq\varepsilon^2 d_v^{-2}/16}} \mathrm{wt}(e) + \sum_{\substack{e\in B \\ \mathrm{wt}e\geq\varepsilon^2 d_v^{-2}/16}} \mathrm{wt}(e) \\
&\leq |B| \times \varepsilon^2 d_v^{-2}/16 + k d_v^{-2} \\
&\leq d_v^2 \times \varepsilon^2 d_v^{-2}/16 + k d_v^{-2} \\
&= \varepsilon^2/16 + k d_v^{-2}.
\end{aligned}$$

Rearranging, $k \geq \varepsilon^2 d_v^2/16$.

Hence, there are at least $\varepsilon^2 d_v^2/16$ edges contained in X with weight at least $\varepsilon^2 d_v^{-2}/16$. Consider the random variable Z that is the weight of a uniform random edge contained in X. Since $|X| \leq \varepsilon^{-5}144d_v$, the number of edges in X is at most $\varepsilon^{-10}(144)^2 d_v^2$. So,

$$\mathbf{E}[Z] \geq \frac{\varepsilon^2 d_v^2/16}{\varepsilon^{-10}(144)^2 d_v^2} \times \varepsilon^2 d_v^{-2}/16 \geq 2\delta\varepsilon^{14} d_v^{-2}. \tag{17}$$

The maximum value of Z is the largest possible weight of an edge in $E(H)$, which is at most d_v^{-2}. Applying the reverse Markov bound of Lemma 3.8, $\Pr[Z \geq \delta\varepsilon^{14} d_v^{-2}] \geq \delta\varepsilon^{14}$. Thus, an ε^{14} fraction of edges in $|X|$ have weight at least $\delta\varepsilon^{14} d_v^{-2} \geq \delta\varepsilon^c/|X|^2$. Moreover, every edge has weight at most $d_v^{-2} \leq 1/(\delta\varepsilon^c|X|^2)$. So we prove the uniformity of $\mathcal{A}|_X$.

The largest possible weight for any edge in $E(H)$ is d_v^{-2}. The size of $|X|$ is at least d_v and at most $\varepsilon^{-5}144d_v$. Hence, $\mathcal{A}|_X$ is at least $\delta\varepsilon^{12}$-uniform.

Proof of strong uniformity: For strong uniformity, we need to repeat the above argument within neighborhoods in X. We prove in the beginning of this proof that the total triangle weight inside X is at least $\varepsilon^3/16$. We also proved that $|X| \leq 146\varepsilon^{-5}d_v$. Consider the random variable Z that is the triangle weight contained in X incident to a uniform random vertex in X. Note that $\mathbf{E}[Z] \geq (\varepsilon^3/16)/(146\varepsilon^{-5}d_v) \geq 2\delta'\varepsilon^8 d_v^{-1}$. By Claim 4.9, Z is at most $(2d_v)^{-1}$. Applying Lemma 3.8, $\Pr[Z \geq \delta'\varepsilon^8 d_v^{-1}] \geq \delta\varepsilon^8$. This means that at least $\delta'\varepsilon^8|X|$ vertices in X are incident to at least $\delta'\varepsilon^8 d_v^{-1}$ triangle weight inside X.

Consider any such vertex u. Let $N(u)$ be the neighborhood of u in X. Every edge e in $N(u)$ forms a triangle with u with weight $\mathrm{wt}(e)/d_u$. Hence, noting that $d_u \geq d_v$,

$$\sum_{e \subseteq N(u)} \mathrm{wt}(e) d_u^{-1} \geq \delta'\varepsilon^8 d_v^{-1} \quad \Longrightarrow \quad \sum_{e \subseteq N(u)} \mathrm{wt}(e) \geq \delta'\varepsilon^8. \tag{18}$$

There are at most $|X|^2 \leq \varepsilon^{-10}(146)^2 d_v^2$ edges in $N(u)$. Let Z denote the weight of a uniform random edge in $N(u)$. Note that $\mathbf{E}[Z] \geq \delta'\varepsilon^8/(\varepsilon^{-10}(146)^2 d_v^2) \geq 2\delta\varepsilon^{18} d_v^{-2}$. The maximum weight of an edge is at most d_v^{-2}. By Lemma 3.8, at least $\delta\varepsilon^{18}$ fraction of edges in $N(u)$ have a weight of at least $\delta\varepsilon^{18} d_v^{-2}$. Since $|N(u)| \leq |X| \leq \varepsilon^{-5}146d_v$, this implies that $N(u)$ is also $\delta\varepsilon^c$-uniform. Hence, we prove strong uniformity as well.

◀

5 Obtaining the decomposition

Algorithm 2 `Decompose`(G)

1: Initialize $\boldsymbol{X}$ to be an empty family of sets, and initialize subgraph $H = G$.
2: **while** H is non-empty **do**
3:     **while** H is not clean **do**
4:         Remove an edge $e \in E(H)$ from H such that $\mathrm{wt}(T_H(e)) < (\varepsilon)\mathrm{wt}(e)$.
5:     **end while**
6:     Add output `Extract`(H) to $\boldsymbol{X}$.
7:     Remove these vertices from H.
8: **end while**
9: Output $\boldsymbol{X}$.

We first describe the algorithm that obtains the decomposition promised in Theorem 1.4.

We partition all the triangles of G into three sets depending on how they are affected by `Decompose`(G). (i) The set of triangles removed by the cleaning step of Step 4, (ii) the set of triangles contained in some $X_i \in \boldsymbol{X}$, or (iii) the remaining triangles. Abusing notation, we refer to these sets as T_C, T_X, and T_R respectively. Note that the triangles of T_R are the triangles "cut" when X_i is removed.

▷ Claim 5.1. $\mathrm{wt}(T_C) \leq (\tau/6)\sum_{e \in E} \mathrm{wt}(e)$.

Proof. Consider an edge e removed at Step 4 of `Decompose`. Recall that ε is set to $\tau/6$. At that removal, the total weight of triangles removed (cleaned) is at most $(\tau/6)\mathrm{wt}(e)$. An edge can be removed at most once, so the total weight of triangles removed by cleaning is at most $(\tau/6)\sum_{e\in E}\mathrm{wt}(e)$. ◀

Proof. (of Theorem 1.4) Let us denote by $H_1, H_2, \dots, H_k$ the subgraphs of which `Extract` is called. Let the output of `Extract`(H_i) be denoted X_i. By the uniformity guarantee of Theorem 4.2, each $\mathcal{A}|_{X_i}$ is $\delta\tau^c$-uniform.

It remains to prove the coverage guarantee. We now sum the bound of Theorem 4.2 over all X_i. (For convenience, we expand out ε as $\tau/6$ and let δ' denote a sufficiently small constant.)

$$\sum_{i\leq k}\sum_{t\in T(H),t\subseteq X}\mathrm{wt}(t) \geq (\delta'\tau^8)\sum_{i\leq k}\sum_{t\in T(H),t\cap X\neq\emptyset}\mathrm{wt}(t). \tag{19}$$

The LHS is precisely $\mathrm{wt}(T_X)$. Note that a triangle appears at most once in the double summation in the RHS. That is because if $t\cap X_i \neq \emptyset$, then t is removed when X_i is removed. Since H_i is always clean, the triangles of T_C cannot participate in this double summation. Hence, the RHS summation is $\mathrm{wt}(T_X)+\mathrm{wt}(T_R)$ and we deduce that

$$\mathrm{wt}(T_X) \geq \delta'\tau^8(\mathrm{wt}(T_X)+\mathrm{wt}(T_R)) \tag{20}$$

Note that $\mathrm{wt}(T_c)+\mathrm{wt}(T_x)+\mathrm{wt}(T_r) = \sum_{t\in T}\mathrm{wt}(t)$. There is where the definition of τ makes its appearance. By Lemma 3.5, we can write the above equality as $\mathrm{wt}(T_c)+\mathrm{wt}(T_x)+\mathrm{wt}(T_r) = (\tau/3)\sum_{e\in E}\mathrm{wt}(e)$. Applying Claim 5.1, (20), and the relation of edge weights to the Frobenius norm (Claim 3.2),

$$(\delta'\tau^8)^{-1}\mathrm{wt}(T_X) \geq (\tau/6)\sum_{e\in E}\mathrm{wt}(e) \quad\Longrightarrow\quad \mathrm{wt}(T_X)\geq \delta\tau^c\|\mathcal{A}\|_2^2 \quad \text{(by Claim 3.2)} \tag{21}$$

By Claim 3.4, $\sum_{i\leq k}\|\mathcal{A}|_{X_i}\|_2^2 \geq \mathrm{wt}(T_X)$, completing the proof of the coverage bound. ◀

6 Algorithmics and implementation

We discuss theoretical and practical implementations of the procedures computing the decomposition of Theorem 1.4. The main operation required is a triangle enumeration of G; there is a rich history of algorithms for this problem. The best known bound for sparse graph is the classic algorithm of Chiba-Nishizeki that enumerates all triangles in $O(m\alpha)$ time, where α is the graph degeneracy.

We first provide a formal theorem providing a running time bound. We do not explicitly describe the implementation through pseudocode, and instead explain the main details in the proof.

▶ **Theorem 6.1.** *There is an implementation of* `Decompose`(G) *whose running time is* $O(R + (m+n+T)\log n)$, *where R is the running time of listing all triangles. The space required is $O(T)$ (where T is the triangle count).*

Proof. We assume an adjacency list representation where each list is stored in a dictionary data structure with logarithmic time operations (like a self-balancing binary tree).

We prepare the following data structure that maintains information about the current subgraph H. We initially set $H = G$. We will maintain all lists as hash tables so that elementary operations on them (insert, delete, find) can be done in $O(1)$ time.

1. A list of all triangles in $T(H)$ indexed by edges. Given an edge e, we can access a list of triangles in $T(H)$ containing e.
2. A list of $\mathrm{wt}(T_H(e))$ values for all edges $e \in E(H)$.
3. A list U of all (unclean) edges such that $\mathrm{wt}(T_H(e)) < \varepsilon \mathrm{wt}(e)$.
4. A min priority queue Q storing all vertices in $V(H)$ keyed by degree d_v. We will assume pointers from v to the corresponding node in Q.

These data structures can be initialized by enumerating all triangles, indexing them, and preparing all the lists. This can be done in $O(R)$ time.

We describe the process to remove an edge from H. When edge e is removed, we go over all the triangles in $T(H)$ containing e. For each such triangle t and edge $e' \in t$, we remove t from the triangle list of e'. We then update $\mathrm{wt}(T_H(e'))$ by reducing it by $\mathrm{wt}(t)$. If $\mathrm{wt}(T_H(e'))$ is less than $\mathrm{wt}(e)$, we add it to U. Finally, if the removal of e removes a vertex v from $V(H)$, we remove v from the priority queue Q. Thus, we can maintain the data structures. The running time is $O(|T_H(e)|)$ plus an additional $\log n$ for potentially updating Q. The total running time for all edge deletes is $O(T + n \log n)$.

With this setup in place, we discuss how to implement `Decompose`. The cleaning operation in `Decompose` can be implemented by repeatedly deleting edges from the list U, until it is empty.

We now discuss how to implement `Extract`. We will maintain a max priority queue R maintaining the values $\{\rho_w\}$. Using Q as defined earlier, we can find the vertex v of minimum degree. By traversing its adjacency list in H, we can find the set L. We determine all edges in L by traversing the adjacency lists of all vertices in L. For each such edge e, we enumerate all triangles in H containing e. For each such triangle t and $w \in t$, we will update the value of ρ_w in R.

We now have the total $\sum_w \rho_w$ as well. We find the sweep cut by repeatedly deleting from the max priority queue R, until the sum of ρ_w values is at least half the total. Thus, we can compute the set X to be extracted. The running time is $O((|X| + |E(X)| + |T(X)|) \log n)$, where $E(X), T(X)$ are the set of edges and triangles incident to X.

Overall, the total time for all the extractions and resulting edge removals is $O((n + m + T) \log n)$. The initial triangle enumeration takes R time. We add to complete the proof. ◀

References

1 N Alon and V.D Milman. λ_1, isoperimetric inequalities for graphs, and superconcentrators. *Journal of Combinatorial Theory, Series B*, 38(1):73–88, 1985.

2 Reid Andersen, Fan Chung, and Kevin Lang. Local graph partitioning using pagerank vectors. In *Foundations of Computer Science (FOCS)*, pages 475–486, 2006.

3 A. Benson, D. F. Gleich, and J. Leskovec. Higher-order organization of complex networks. *Science*, 353(6295):163–166, 2016.

4 Ronald S. Burt. Structural holes and good ideas. *American Journal of Sociology*, 110(2):349–399, 2004.

5 Deepayan Chakrabarti and Christos Faloutsos. Graph mining: Laws, generators, and algorithms. *ACM Comput. Surv.*, 38(1):2–es, jun 2006.

6 Jeff Cheeger. *A Lower Bound for the Smallest Eigenvalue of the Laplacian*, pages 195–200. Princeton University Press, 1971.

7 F. R. K. Chung. *Spectral Graph Theory.* American Mathematical Society, 1997.

8 Katherine Faust. Comparing social networks: Size, density, and local structure. *Metodoloski zvezki*, 3:185–216, 07 2006.

9 M. Fiedler. Algebraic connectivity of graphs. *Czechoslovak Mathematical Journal*, 23(98):298–305, 1973.

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

11 Santo Fortunato and Mark E. J. Newman. 20 years of network community detection. *Nature Physics*, 18(8):848–850, jul 2022.

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

13 D. Gleich and C. Seshadhri. Vertex neighborhoods, low conductance cuts, and good seeds for local community methods. In *SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pages 597–605, 2012.

14 M. Granovetter. The strength of weak ties: A network theory revisited. *Sociological Theory*, 1:201–233, 1983.

15 Rishi Gupta, Tim Roughgarden, and C. Seshadhri. Decompositions of triangle-dense graphs. *Innovations in Theoretical Computer Science*, pages 471–482, 2014.

16 P. Holland and S. Leinhardt. A method for detecting structure in sociometric data. *American Journal of Sociology*, 76:492–513, 1970.

17 Paul W. Holland and Samuel Leinhardt. Local structure in social networks. *Sociological Methodology*, 7:1–45, 1976.

18 Mark Jerrum and Alistair Sinclair. Approximating the permanent. *SIAM Journal on Computing*, 18(6):1149–1178, 1989.

19 J. Kleinberg. Navigation in a small world. *Nature*, 406(6798), 2000.

20 James R. Lee, Shayan Oveis Gharan, and Luca Trevisan. Multiway spectral partitioning and higher-order cheeger inequalities. *J. ACM*, 61(6):1–30, 2014.

21 J. Leskovec, K. J. Lang, A. Dasgupta, and M. W. Mahoney. Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters. *Internet Mathematics*, 6(1):29–123, 2009.

22 S. Milgram. The small world problem. *Psychology Today*, 1(1):60–67, 1967.

23 M. E. J. Newman. Properties of highly clustered networks. *Phys. Rev. E*, 68:026121, Aug 2003.

24 M. E. J. Newman. Finding community structure in networks using the eigenvectors of matrices. *Phys. Rev. E*, 74:036104, Sep 2006.

25 M. E. J. Newman. Modularity and community structure in networks. *Proceedings of the National Academy of Sciences*, 103:8577–8582, 2006.

26 Andrew Y. Ng, Michael I. Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. In *Neural Information Processing Systems: Natural and Synthetic*, NIPS'01, page 849–856. MIT Press, 2001.

27 Richard Peng, He Sun, and Luca Zanetti. Partitioning well-clustered graphs: Spectral clustering works! In *Conference on Learning Theory (COLT)*, volume 40 of *Proceedings of Machine Learning Research*, pages 1423–1455, 2015.

28 A. Erdem Sariyuce, C. Seshadhri, A. Pinar, and U. Catalyurek. Finding the hierarchy of dense subgraphs using nucleus decompositions. In *World Wide Web (WWW)*, pages 927–937, 2015.

29 C. Seshadhri, Tamara G. Kolda, and Ali Pinar. Community structure and scale-free collections of Erdos-Renyi graphs. *Physical Review E*, 85:056109, 2012.

30 Daniel A. Spielman. *Spectral and Algebraic Graph Theory.* `http://cs-www.cs.yale.edu/homes/spielman/sagt/`.

31 Daniel A. Spielman. Spectral graph theory and its applications. In *IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 29–38, 2007.

32 Daniel A. Spielman and Shang-Hua Teng. A local clustering algorithm for massive graphs and its application to nearly linear time graph partitioning. *SIAM Journal on Computing*, 42(1):1–26, 2013.

33 Charalampos E. Tsourakakis. The k-clique densest subgraph problem. In *The Web Conference (WWW)*, pages 1122–1132, 2015.

34 Charalampos E. Tsourakakis, Jakub Pachocki, and Michael Mitzenmacher. Scalable motif-aware graph clustering. In *The Web Conference (WWW)*, pages 1451–1460, 2017.
35 Stanley Wasserman and Katherine Faust. *Social Network Analysis: Methods and Applications*. Structural Analysis in the Social Sciences. Cambridge University Press, 1994.
36 D. Watts and S. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393:440–442, 1998.
37 Zeyuan Allen Zhu, Silvio Lattanzi, and Vahab S. Mirrokni. A local algorithm for finding well-connected clusters. In *International Conference on Machine Learning (ICML)*, volume 28, pages 396–404, 2013.