Signed Laplacian for Spectral Clustering Revisited

Knyazev, A.

TR2017-001   January 2017

Abstract

Classical spectral clustering is based on a spectral decomposition of a graph Laplacian, obtained from a graph adjacency matrix representing positive graph edge weights describing similarities of graph vertices. In signed graphs, the graph edge weights can be negative to describe disparities of graph vertices, for example, negative correlations in the data. Negative weights lead to possible negative spectrum of the standard graph Laplacian, which is cured by defining a signed Laplacian. We revisit comparing the standard and signed Laplacians and argue that the former is more natural than the latter, also showing that the negative spectrum is actually beneficial, for spectral clustering of signed graphs.

arXiv

SIGNED LAPLACIAN FOR SPECTRAL CLUSTERING REVISITED[*]

ANDREW V. KNYAZEV[§][¶][‖]

Abstract. Classical spectral clustering is based on a spectral decomposition of a graph Laplacian, obtained from a graph adjacency matrix representing positive graph edge weights describing similarities of graph vertices. In signed graphs, the graph edge weights can be negative to describe disparities of graph vertices, for example, negative correlations in the data. Negative weights lead to possible negative spectrum of the standard graph Laplacian, which is cured by defining a signed Laplacian. We revisit comparing the standard and signed Laplacians and argue that the former is more natural than the latter, also showing that the negative spectrum is actually beneficial, for spectral clustering of signed graphs.

Key words. spectral clustering, signed graph,signed Laplacian, mass-spring vibration

AMS subject classification. 05C50, 05C70, 15A18, 58C40, 65F15, 65N25, 62H30, 91C20.

1. Background and Motivation. Spectral clustering groups together related data points and separates unrelated data points, using spectral graph theory, i.e., spectral properties of matrices associated with the graph, such as graph adjacency and Laplacian matrices; see, e.g., [23, 24, 27, 28, 30, 33, 37]. Internet search for "spectral clustering" nowadays returns over 200,000 links, 25,000 scholar publications, and 1,000 patent applications.

The graph Laplacian matrix is obtained from the graph adjacency matrix that represents graph edge weights describing similarities of graph vertices. The graph weights are commonly defined using a distance function measuring similarity between data points, where graph vertices represent the data points and graph edges are drawn between pairs of vertices, e.g., if the distance between the corresponding data points has been measured. Classical spectral clustering bi-partitions the graph according to the signs of the components of the Fiedler vector; see [7, 8].

Some important applications, e.g., Slashdot Zoo [20] and correlation [2] clustering, naturally lead to signed graphs, i.e., both with positive and *negative* weights. Negative values in the graph adjacency matrix result in more difficult spectral graph theory; see, e.g., [3]. Applying the original definition of the graph Laplacian to signed graphs breaks many useful properties of the graph Laplacian, e.g., leading to negative eigenvalues, making the definition of the Fiedler vector ambivalent. The row-sums of the graph Laplacian may vanish, invalidating the definition of the normalized Laplacian.

These difficulties are typically avoided by using the *signed Laplacian*, e.g., [9, 19, 21], defined similarly to the graph Laplacian, but with the diagonal entries forced to be always positive large enough to make the signed Laplacian positive semi-definite and its eigenvalues non-negative, allowing one to formally reuse the definitions of the Fiedler vector and the normalized Laplacian.

In this work, we argue that the original graph Laplacian is a more natural and beneficial choice, compared to the signed Laplacian, for spectral clustering. Using a model of vibrations of a mass-spring system with some repulsive springs, we explain, why the definition of the Fiedler vector should be based on the smallest eigenvalue,

no matter whether it is positive or negative. In comparison, an "inverting amplifier" idea in [21, Sec. 7] appears artificial, lacks intuition and leads to a questionable argument, where the sign of negative edges changes in the denominator, but not in the numerator, of the potential. Turning to justification of the spectral clustering via relaxation, we compare the standard "ratio cut," e.g., [27, 28], and "signed ratio cut" of [21], noting that minimizing the signed ratio cut puts less emphasis on cutting negative edges. We illustrate the behavior of the Fiedler vector for an intuitively trivial case of partitioning a linear graph modelled by vibrations of a string. We demonstrate numerically and analyze deficiencies of the popular signed Laplacian vs. the standard Laplacian for spectral clustering on a few simple examples.

Efficiency of solvers for eigenvalue problems in spectral clustering is crucial for practical computations, where the data become enormous in size and multi-threaded computational platforms get easily available; e.g. cf. [1, 6, 18, 26, 36].

A closely related to spectral clustering area of research is graph-based signal processing, that introduces eigenvectors of the graph Laplacian as natural extensions of the Fourier bases. For example, in [15], graph-based edge-preserving denoising is performed by projecting the noisy signal on a lower dimensional Krylov subspace of the graph Laplacian, constructed using the signal. The construction of the graph Laplacian of [15] is extended in [14] to the case, where weights can be negative, leading to edge enhancing denoising. In the present work, we extend this idea to clustering.

The rest of the paper is organized as follows. We formally introduce the spectral clustering in Section 2 via eigendecomposition of the graph Laplacian. Section 3 deals with a simple, but representative, example—a linear graph,—and motivates spectral clustering by utilizing properties of low frequency mechanical vibration eigenmodes of a string, as an example of a mass-spring model. Negative edge weights are then naturally introduced in Section 4 as corresponding to repulsive springs, and the effects of the negative weights on the eigenvectors of the Laplacian are informally predicted by the repulsion of the masses connected by the repulsive spring. In Section 5, we present simple motivating examples, discuss how the original and signed Laplacians are introduced via relaxation of combinatorial optimization, and numerically compare their eigenvectors and gaps in the spectra. In Sections 6 and 7, we explain benefits and peculiarities of using the negative weights for spectral clustering. Possible future research directions are spotlighted in Section 8.

2. Brief introduction to spectral clustering. Let entries of the symmetric N-by-N *data similarity* matrix matrix W be called *weighs* and the matrix D be diagonal, made of row-sums of the matrix W. The matrix W is a data similarity matrix that may be viewed as a matrix of scores, representing the similarity between pairs of N data points. Similarity matrices are commonly determined from their counterparts, distance matrices. The distance matrix is a matrix containing the distances, taken pairwise, of a set of points. The general connection is that the similarity is small if the distance is large, and vice versa. Traditionally, all the entries (weighs) in W are assumed to be non-negative, which is automatically satisfied for distance-based similarities. We extend clustering to a general case of both positive and negative weighs, e.g., associated with correlations between pairs of data points.

Commonly, the data clustering problem is formulated as a graph partition problem, defined on data represented in the form of a graph $G = (V, E)$, with N vertices V and M edges E such that it is possible to partition G into sub-graphs. Entries of W are weighs of the corresponding edges E, and the matrix W is called the *graph adjacency* matrix. The graph is called *signed* if it has edges with negative weighs.

Let us introduce the Laplacian $L = D - W$ and the normalized Laplacian $D^{-1}L = I - D^{-1}W$ matrices, assuming that D is invertible, where I is the identity matrix. Normalizing the graph Laplacian is basically vertex scaling, conveniently mapping the spectrum of the normalized Laplacian $D^{-1}L$ on the real interval $[0, 2]$, if the edge-weighs are all non-negative. To simplify our presentation for the signed graphs, we mostly avoid the normalized Laplacian.

A good partition in the classical case of non-weighted graphs is informally described as the one in which the number of edges between separated sub-graphs is small. The goal of partitioning of signed graphs, e.g., by cutting into two sub-graphs, is informally to minimize the total weight of the cut positive edges, while at the same time to maximize the absolute total weight of the cut negative edges. For uniform partitioning, one also needs to well-balance sizes or volumes of the resulting sub-graphs. Traditional approaches to graph partitioning are combinatorial and naturally fall under the category of NP-hard problems, solved using heuristics, such as relaxing the combinatorial constraints, as we briefly discuss below in Subsection 5.2.

Data clustering via graph spectral partitioning is a state-of-the-art tool, which is known to produce high quality clusters at reasonable costs of numerical solution of an eigenvalue problem for a matrix associated with the graph, e.g., the graph Laplacian. Each eigenvector, denoted by x, of the matrix L is a solution of the eigenvalue problem $Lx = \lambda x$, where the scalar λ denotes the eigenvalue corresponding to the eigenvector x. Together, λ and x are called an *eigenpair*.

The column-vector of ones is always a trivial eigenvector of L corresponding to the zero eigenvalue. It is easy to show that the symmetric matrix L has all non-negative eigenvalues λ, if all weights are non-negative. The graph Laplacian matrix is symmetric, due to the earlier assumed symmetry of the adjacency matrix W, which implies that all the eigenvalues are real, and the various eigenvectors can be chosen to be mutually orthogonal. A nontrivial eigenvector of the matrix L corresponding to smallest eigenvalue λ of L, commonly called the Fiedler vector after the author of [7, 8], bisects the graph G into only two partitions, according to the signs of the entries of the eigenvector. Since the Fielder vector, as any other nontrivial eigenvector, is orthogonal to the vector of ones, it must have entries of opposite signs, thus, the sign-based bisection always generates a non-trivial two-way graph partitioning. We explain in Section 3, why such a partitioning method can be considered meaningful.

A multiway spectral partitioning is obtained from "low frequency eigenmodes," i.e., eigenvectors corresponding to a cluster of smallest eigenvalues, of the Laplacian matrix L. The cluster of (nearly)-multiple eigenvalues naturally leads to the need of considering a Fiedler invariant subspace of L, spanned by the corresponding eigenvectors, extending the Fiedler vector, since the latter may be not unique or well numerically defined in this case. The Fiedler invariant subspace provides a geometric embedding of the graph's vertices, reducing the general graph clustering problem to the problem of clustering of a point cloud (of embedded graph vertices) in a low-dimensional Euclidean space. However, the simple and efficient idea of sign-based partitioning from the Fiedler vector has no evident extension to the Fiedler invariant subspace. Practical multiway spectral partitioning can be performed using various competing heuristic algorithms, greatly affecting the results; see, e.g., [5, 12, 31, 35]. While these same heuristic algorithms can as well be used in our context of signed graphs, we restrict ourselves in this work only to the two-way partitioning using the component signs of the Fiedler vector, for clarity of presentation and consistency.

The presence of negative weights in signed graphs brings new theoretical and practical challenges to spectral graph partitioning:

- possible negative eigenvalues of the graph Laplacian, making the definition of the Fiedler vector somewhat ambiguous, e.g., whether the smallest negative, positive, or by absolute value, eigenvalues should be used in the definition;
- more difficult spectral graph theory; e.g., cf. [24] and [9];
- possible zero diagonal entries in the degree matrix D, nullifying the definition of the normalized Laplacian $D^{-1}L$, cf. [33];
- violation of the maximum principle, which is the cornerstone in the proof of connectivity of clusters; see [7, 8];
- breaking the connection of spectral clustering to random walks and Markov chains, cf. [27];
- invalidating an interpretation of the quadratic form $x^\mathsf{T} L x$ as "energy," e.g., in the heat (diffusion) equation; cf. a forward-and-backward diffusion in [10, 34];
- the graph Laplacian can no longer be viewed as a discrete analog of the Laplace-Beltrami operator on a Riemannian manifold that motivates Laplacian Eigenmaps methods for manifold learning; e.g., see [11, 32].

Some of these challenges can be addressed by defining a *signed* Laplacian as follows. Let the matrix $\bar{D}$ be diagonal, made of row-sums of the *absolute values of the entries* of the matrix W, which thus are positive, so that $\bar{D}^{-1}$ is well-defined. We define the *signed* Laplacian $\bar{L} = \bar{D} - W$ and the *normalized signed* Laplacian $\bar{D}^{-1}\bar{L} = I - \bar{D}^{-1}W$ matrices, following, e.g., [9, 19, 21]. The signed Laplacian is positive semi-definite, with all eigenvalues non-negative. The Fiedler vector of the signed Laplacian is defined in [9, 19, 21] as an eigenvector corresponding to the smallest eigenvalue and different from the trivial constant vector.

In the rest of the paper, we justify spectral partitioning of signed graphs using the original definition of the graph Laplacian L, and argue that better quality clusters can generally be expected from eigenvectors of the original L, rather than from the signed Laplacian $\bar{L}$.

3. Linear graph Laplacian and low frequency eigenmodes of a string.

3.1. All edges with unit weights. Spectral clustering can be intuitively justified via a well-known, see, e.g., [29], identification of the graph Laplacian matrix L with a classical problem of transversal vibrations of a mass-spring system with N masses and M springs, where the stiffness of the springs is related to the graph weights—the entries of the graph adjacency matrix W. We start with the simplest non-trivial example, where the mass-spring system is a discrete analog of a string.

Let $w_{i-1\,i} = w_{i\,i+1} = 1$ with all other zero entries, so that the graph Laplacian $L = D - W$ is a tridiagonal matrix

$$(1) \qquad L = \begin{pmatrix} 1 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 1 \end{pmatrix}$$

that has nonzero entries 1 and -1 in the first row, -1 and 1 in the last row, and $[-1\ \ 2\ \ -1]$ in every other row—a standard finite-difference approximation of the negative second derivative of functions with vanishing first derivatives at the end points of the interval. Its eigenvectors are the basis vectors of the discrete cosine transform; see the first five low frequency eigenmodes (the eigenvectors corresponding to the smallest eigenvalues) of L displayed in the left panel in Figure 1.
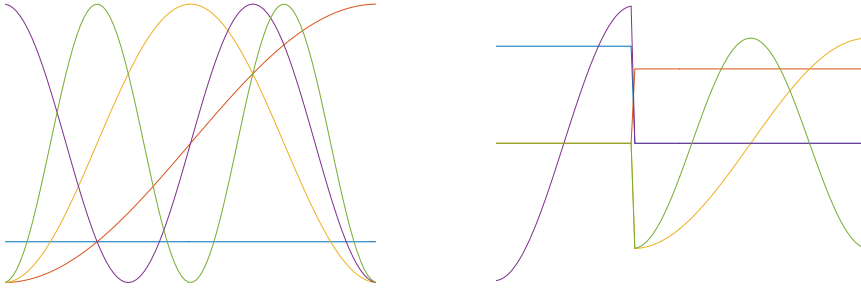
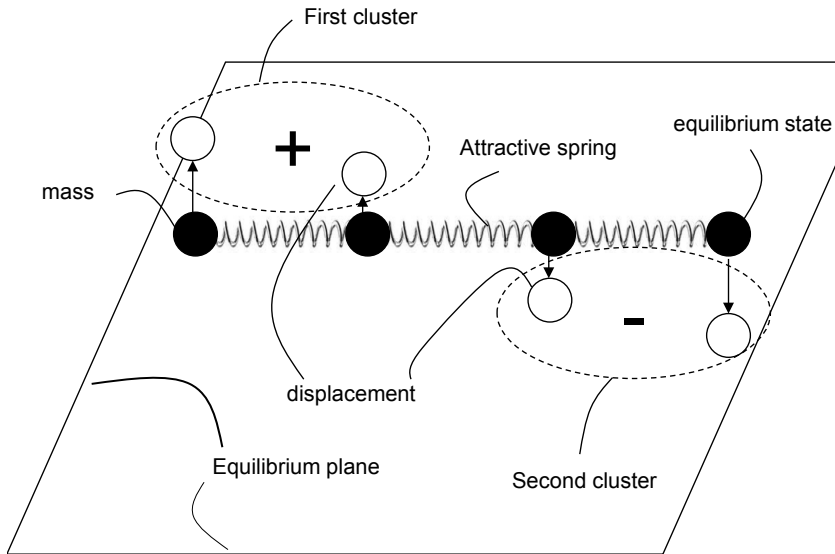FIG. 1. *Low frequency eigenmodes of a string (left) and two disconnected pieces of the string (right).*

FIG. 2. *Traditional linear mass-spring system.*

Let us note that these eigenmodes all turn flat at the end points of the interval. The flatness is attributed to the vanishing first derivatives, which manifests itself in the fact, e.g., that the Laplacian row sums in the first and last rows always vanish.

Eigenvectors of matrix (1) are well-known in mechanics, as they represent shapes of transversal vibration modes of a discrete analog of a string—a linear system of masses connected with springs. Figure 2 illustrates a system with $N = 4$ masses and $M = 3$ springs.

The frequency ω of the vibration mode x satisfies $\omega^2 = \lambda \geq 0$. The eigenvectors of the graph Laplacian are often called *eigenmodes* because of this mechanical analogy. The smallest eigenvalues $\lambda = \omega^2$ correspond to low frequencies ω, explaining the terminology used in the caption in Figure 1. The system of masses is not attached, thus there is always a trivial eigenmode, where the whole system goes up/down, i.e., the eigenvector x is constant and $\omega^2 = \lambda = 0$.

Figure 2 shows transversal displacements of the masses from the equilibrium plane for the first nontrivial mode, which is the Fiedler vector, where the two masses on the left side of the system move synchronously up, while the two masses on the right side of the system move synchronously down. This is the same eigenmode as drawn in red color in Figure 1 left panel for a similar linear system with a number of masses large enough to make a visual appearance of a continuous string. Performing the bisection partitioning according to the signs of the Fiedler vector, one puts the data points corresponding to the masses in the left half of the mass-spring system into one cluster and those in the right half into the other.

The amplitudes of the Fiedler vector are also very important. For example, the Fiedler vector in Figure 2 has smaller by absolute value components in the middle of the system. With the number of masses increased, the components in the middle of the system gets smaller. Then, a perturbation of the graph weights can easily lead to a sign change in a small by the absolute value component, putting the corresponding data point into a difference cluster, as intuitively should be expected. Thus, the squared amplitude of the component, after a proper scaling, can be interpreted as a probability of the corresponding data point to belong to the cluster determined according to the sign of the component.

3.2. A string with a single weak link (small edge weight). Next, we set a very small value $w_{i\,i+1} = w_{i+1\,i}$ for some index i, keeping all other entries of the matrix W the same as before. In terms of clustering, this example represents a situation, where there is an intuitively evident bisection partitioning, with one cluster containing all data points with indexes $1, \ldots, i$ and the other with $i + 1, \ldots, N$. In terms of our mass-spring system interpretation, we have a discrete string with one weak link, i.e., one spring with such a small stiffness that makes two pieces of the string nearly disconnected. Let us check how the low frequency eigenmodes react to such a change.

The first five vectors of the corresponding Laplacian are shown in Figure 1, right panel. We observe that all the plotted in Figure 1 eigenvectors are aware of the soft (having small stiffness) spring between the masses with the indexes i and $i + 1$. Moreover, their behavior around the soft spring is very specific—they are all flat on both sides of the soft spring!

The presence of the flatness in the low frequency modes of the graph Laplacian L on both sides of the soft spring is easy to explain mathematically. When the value $w_{i\,i+1} = w_{i+1\,i}$ is small relative to other entries, the matrix L becomes nearly block diagonal, with two blocks, which approximate graph Laplacian matrices on sub-strings to the left and to the right of the soft spring.

The low frequency eigenmodes of the graph Laplacian L thus approximate combinations of the low frequency eigenmodes of the graph Laplacians on the sub-intervals. But each of the low frequency eigenmode of the graph Laplacian on the sub-interval is flat on both ends of the sub-interval, as explained above. Combined, it results in the flatness in the low frequency modes of the graph Laplacian L on both sides of the soft spring.

The flatness is also easy to explain in terms of mechanical vibrations. The soft spring between the masses with the indexes i and $i + 1$ makes the masses nearly disconnected, so the system can be well approximated by two independent disconnected discrete strings with free boundary conditions, on the left and on the right to the soft spring. Thus, the low frequency vibration modes of the system are discontinuous at the soft spring, and nearly flat on both sides of the soft spring.
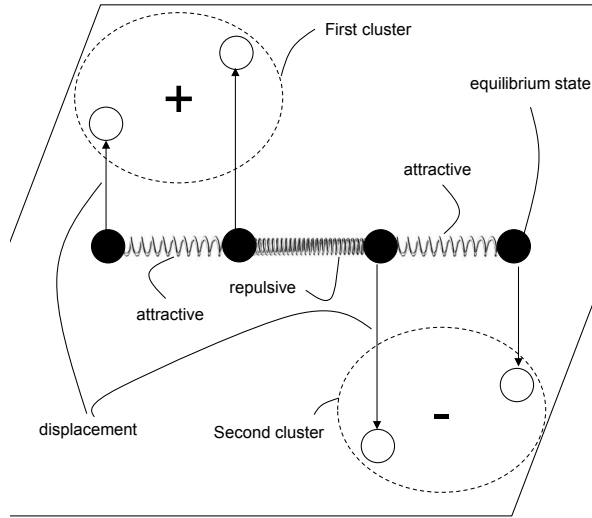
FIG. 3. *Linear mass-spring system with repulsion.*

Can we do better in this toy example and make the flat ends to bend in the opposite directions, making it easier to determine the bisection partitioning even using low accuracy computations of the eigenvectors? In [14], where graph based edge-preserving signal denoising is analyzed, we have suggested to enhance the edges of the signal by introducing negative edge weights in the graph, cf. [10]. In the next subsection, we put a spring, which separates the masses by repulsing them, and see how it affects the low-frequency vibration modes.

4. Negative weights for spectral clustering. In our mechanical vibration model of a spring-mass system, the masses that are tightly connected have a tendency to move synchronically in low-frequency free vibrations. Analyzing the signs of the components corresponding to different masses of the low-frequency vibration modes thus allows determining the clusters.

The mechanical vibration model describes conventional clustering when all the springs are pre-tensed to create attracting forces between the masses, where the mass-spring system is subject to transverse vibrations, i.e., enable the masses to move only in a transverse direction, perpendicular to a plane of the mass-spring system. However, one can also pre-tense some of the springs to create repulsive forces between some pairs of masses, as illustrated in Figure 3. In the context of data clustering mathematically formulated as graph partitioning, that corresponds to negative entries in the adjacency matrix. The negative entries in the adjacency matrix are not allowed in conventional spectral graph partitioning. However, the model of mechanical vibrations of the spring-mass system with repulsive springs is still valid, allowing us to extend the conventional approach to the case of negative weights.

The masses, which are attracted, move together in the same direction in low-frequency free vibrations, while the masses, which are repulsed, have the tendency to move in the opposite direction. Moreover, the eigenmode vibrations of the spring-mass system relate to the corresponding wave equation, where the repulsive phenomenon makes it possible for the time-dependent solutions of the wave equation to exponentially grow in time, if they correspond to negative eigenvalues.
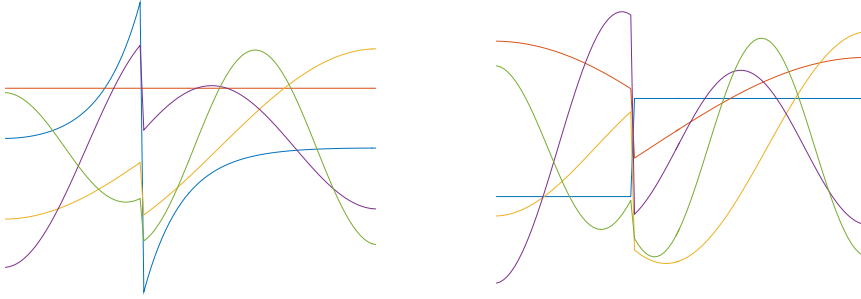
Fig. 4. *The same eigenmodes, but negative weights, original (left) and signed (right) Laplacians.*

Figure 3 shows the same linear mass-spring system as Figure 2, except that the middle spring is repulsive, bending the shape of the Fiedler vector in the opposite directions on different sides of the repulsive spring. The clusters in Figure 2 and Figure 3 are the same, based on the signs of the Fiedler vectors. However, the data points corresponding to the middle masses being repulsed more clearly belong to different clusters in Figure 3, compared to Figure 2, because the corresponding components in the Fiedler vector are larger by absolute value in Figure 3 vs. Figure 2. Determination of the clusters, using the signs of the Fiedler vector, is easier for larger components that are less likely to be computed with a wrong sign due to data noise or inaccuracy of computations, e.g., small number of iterations.

Figures 4 left panel displays the five eigenvectors, including the trivial one, for the five smallest eigenvalues of the same tridiagonal graph Laplacian as that corresponding to the right panel in Figure 1 except that the small positive entry of the weights $w_{i\,i+1} = w_{i+1\,i}$ for the same i is substituted by -0.05 in Figure 4.

5. Comparing the original and signed Laplacians. We present a few simple motivating examples, discuss how the original and signed Laplacians are introduced via relaxation of combinatorial optimization, and compare their eigenvectors and gaps in the spectra, computed numerically for these examples.

5.1. Simple examples. The graph corresponding to the mass-spring system in Figure 5, assuming unit by absolute value weights of the edges, leads to the following graph Laplacian

$$(2) \qquad L = \begin{pmatrix} -2 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 \\ 1 & 0 & 0 & -1 & 0 \\ 0 & -1 & -1 & 3 & -1 \\ 0 & 0 & 0 & -1 & 1 \end{pmatrix},$$

and the signed Laplacian

$$(3) \qquad \bar{L} = \begin{pmatrix} 2 & 1 & 1 & 0 & 0 \\ 1 & 2 & 0 & -1 & 0 \\ 1 & 0 & 2 & -1 & 0 \\ 0 & -1 & -1 & 3 & -1 \\ 0 & 0 & 0 & -1 & 1 \end{pmatrix},$$
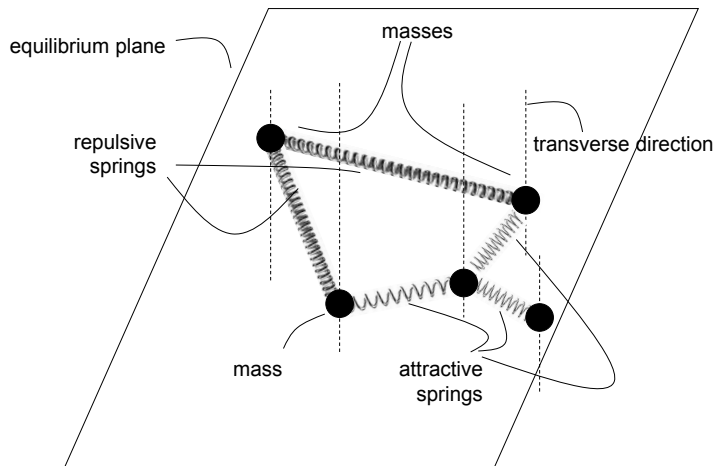
FIG. 5. *Mass-spring system with repulsive springs.*

where the top left in Figure 5 mass, positioned between the repulsive springs, is assigned the index 1, i.e., corresponds to the first row and column of L. Even without computing the eigenvalue decomposition, one can easily predict that the eigenvector of L, orthogonal to the trivial eigenvector of ones and corresponding to the smallest eigenvalue, should have the first component of the opposite sign compared to all other components, separating the first mass from the rest.

To setup a proper comparison, let us consider a graph like the one corresponding to the mass-spring system in Figure 5, but with both repulsive springs being already eliminated. In terms of the graph, we substitute both negative weights with the zero weights, leading to the graph Laplacian

$$
(4) \qquad L_0 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 1 & -1 & 0 \\ 0 & -1 & -1 & 3 & -1 \\ 0 & 0 & 0 & -1 & 1 \end{pmatrix}.
$$

The Fiedler vector of L_0 in (4) is evidently piece-wise constant, with the first component having the opposite sign compared to all other components, separating the first mass from the rest. Let x_0 denote the Fiedler vector of L_0 in (4) shifted by the vector of ones and scaled so that its components with the opposite sign are simply $+1$ and -1, i.e., e.g., $x_0 = (-1\,1\,1\,1\,1)^\mathsf{T}$. We still have that $L_0 x_0 = 0$. Considering

$$
\bar{L} - L_0 = \begin{pmatrix} 2 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix},
$$

we get $(\bar{L} - L_0)x_0 = 0$, thus, also $\bar{L}x_0 = 0$, i.e., x_0 is also an eigenvector of $\bar{L}$, corresponding to the smallest eigenvalue. Therefore, in this example, we come to the surprising conclusion that performing graph spectral bi-partitioning based on the

signed Laplacian has no advantages compared to the spectral bi-partitioning, where the negative weights are simply nullified!

In another example, for the discrete string with one negative edge, we obtain

$$\bar{L} - L_0 = \begin{pmatrix} \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & 0 & 1 & 1 & 0 & \cdots \\ \cdots & 0 & 1 & 1 & 0 & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \end{pmatrix},$$

where the graph Laplacian L_0 corresponds to the two disconnected string pieces, and all dotted entries are zeros. Again, we observe that the eigenvector x_0 of L_0 such that $L_0 x_0 = 0$ is piece-wise constant, and can be chosen to have components $+1$ and -1, separating the left string piece from the right one. And yet again, $(\bar{L} - L_0)x_0 = 0$, thus, $\bar{L}x_0 = 0$, i.e., x_0 is an eigenvector of both matrices $\bar{L}$ and L_0, corresponding to the smallest eigenvalue, in this example.

5.2. Spectral clustering via relaxation. A common approach to formulate spectral graph partitioning is via relaxation of combinatorial minimization problems. Let us compare the standard "ratio cut," e.g., [27, 28], leading to the traditional graph Laplacian, and "signed ratio cut" of [21], used to justify the definition of the signed Laplacian. Let the graph V be cut into sub-graphs X and $V \setminus X$. The associated cut value $Cut(X, V \setminus X)$ is defined as the number of cut edges for graphs without weight and the sum of the weights of cut edges for weighted graphs. In signed graphs, thus, $Cut(X, V \setminus X) = Cut^+(X, V \setminus X) - Cut^-(X, V \setminus X)$, where $Cut^+(X, V \setminus X)$ $(Cut^-(X, V \setminus X))$ denotes the sum of the absolute values of the weights of cut positive (negative) edges. For uniform partitioning, one wants to balance the resulting sub-graphs, which is typically done dividing the cut by the size of the clusters, giving $RatioCut(X, V \setminus X) = Cut(X, V \setminus X)/(|X| + |V \setminus X|)$.

Minimizing the RatioCut is a combinatorial problem, but its relaxation, see e.g., [27, 28], leads to minimizing the Rayleigh quotient for the graph Laplacian. Substituting sub-graph volumes for sizes similarly leads to the normalized Laplacian, not covered here.

The signed ratio cut of [21] is defined by substituting the "signed cut"

$$SignedCut(X, V \setminus X) \equiv 2Cut^+(X, V \setminus X) + Cut^-(X, X) + Cut^-(V \setminus X, V \setminus X)$$

for the "cut". However, the value

$$Cut^-(X, V \setminus X) + Cut^-(X, X) + Cut^-(V \setminus X, V \setminus X)$$

of all negative edges in the signed graph remains constant, no matter what X is. We notice that, up to this constant value, $SignedCut(X, V \setminus X)$ is equal to

$$2Cut^+(X, V \setminus X) - Cut^-(X, V \setminus X).$$

This expression is similar to that of $Cut(X, V \setminus X)$, but the term $Cut^+(X, V \setminus X)$ appears with the multiplier 2, which suggests that the cuts minimizing quantities involving $SignedCut(X, V \setminus X)$ could tend to ignore the edges with negative weights, focusing instead on cutting the edges with small positive weights. In contrast, in the definition of $Cut(X, V \setminus X)$, the positive and negative weights play equal roles. It appears hard to argue why minimizing the signed ratio cut would make more sense compared to the standard ratio cut for signed graphs. Moreover, *doubling* the term $Cut^+(X, V \setminus X)$ appears somewhat arbitrary.

While the relaxation argument is often used as a motivation for spectral clustering, it is difficult to mathematically analyze how different cost functions in the combinatorial formulation affect their relaxed versions. It is also challenging to directly quantitively compare various spectral clustering formulations, where the clusters are determined from eigenvectors, since shapes of eigenvectors depend on the matrix coefficients in a complex way. We use the intuitive mass-spring system model to explain novel effects of spring repulsion on eigenmodes of the standard Laplacian, but we are unaware of such a physical model for the signed Laplacian.

5.3. Comparing the eigenvectors. Since there is apparently still no algorithm universally accepted by experts for an ultimate determination of multiway clusters from several eigenvectors, we restrict ourselves to determining the clusters from the component signs of only one eigenvector, the Fiedler vector for the traditional Laplacian, and the first, or the second, smallest eigenvector of the signed Laplacian, as advocated in [21].

The shape of this single eigenvector that determines bi-partitioning is not the only important factor for spectral clustering. Since the eigenvector has to be computed numerically and is subject to data noise, it is also important to check the shapes of other eigenvectors, corresponding to the nearby eigenvalues, that are likely to contaminate the approximate eigenvector of interest.

Figure 4 right panel displays the eigenmodes of the signed Laplacian for the same weights as in the left panel. We observe that, indeed, as we have proved above, one of the eigenvectors is piece-wise constant, as in Figure 1 right panel. Moreover, the shapes of the other eigenmodes of the signed Laplacian in Figure 4 right panel also look more similar to those in Figure 1 right panel, corresponding to the zero weight, than Figure 4 left panel, corresponding to the original graph Laplacian with the same weights. The eigenvectors the original Laplacian with negative weights are preferably shaped, compared to those of the signed Laplacian, potentially giving better clustering.

To setup a direct numerical comparison, for the string example, we need to choose a practical eigensolver, so let us briefly discuss computational aspects of spectral clustering. The Fiedler vector, or a group of eigenvectors, corresponding to the leftmost eigenvalues of a symmetric eigenvalue problem can be computed by an iterative method, for example, as proposed in [16], by the Locally Optimal Block Preconditioned Conjugate Gradient (LOBPCG) method; see [13]. LOBPCG does not even need to store the matrix L in memory, but only needs the results from multiplying the matrix L by a given vector, or a block of vectors. This characteristic makes LOBPCG applicable to eigenvalue analysis problems of very high dimensions, and results in good parallel scalability to large matrix sizes processed on many parallel processors; e.g., see reference [18], describing one open source and publicly available implementation of LOBPCG. Available convergence theory of LOBPCG in [13] requires the matrix be symmetric, but not necessarily with all non-negative eigenvalues, i.e., a possible presence of negative eigenvalues still satisfies the convergence assumptions. The calculation of the product of the matrix L by a vector is the main cost per iteration, no matter if the weights are positive or negative.

To this end, we perform 30 iterations of LOBPCG, without preconditioning and starting from a random initial approximation—the same for various choices of the weights and for different Laplacians for our discrete string example. The number of iterations is chosen small enough to amplify the influence of inaccuracy in approximating the eigenvector on its shape. We display a representative case in Figure 6.
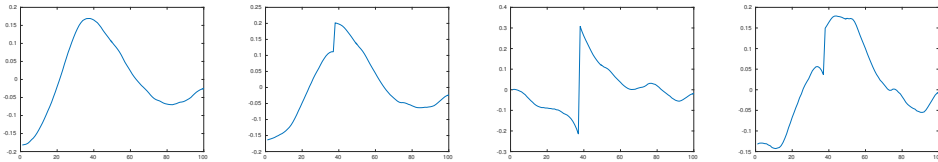
Fig. 6. *Approximate Laplacian eigenmode, unit (a), zero (b), and negative weight at one edge for the original (c) and signed (d) Laplacian.*

Figure 6 shows the approximately computed Laplacian eigenmodes with the unit (a), zero (b), and negative (c) weight at one edge, as well as the signed Laplacian (d), corresponding to the exact eigenfunctions in Figures 1 and 4. Initial large contributions from other eigenmodes, shown in Figures 1 and 4, remain unresolved, as anticipated. If the bi-partitioning was performed according to the signs of the components of the computed eigenmode of the Laplacian with the negative weight nullified, Figure 6 (b), or the signed Laplacian, Figure 6 (d), the resulting clusters would be wrong.

In a sharp contrast, the exact eigenmode of the original Laplacian with the negative weight -0.05, Figure 4 left panel the blue line, as well its approximation in Figure 6 (c), which is very different from all other approximate eigenmodes in Figure 6, clearly demonstrate a sharp edge with a large jump in the functions at the correct location of the negative edge, between the 37 and 38 vertices. The signs of their components allow one to determine the optimal bi-partitioning.

5.4. Comparing gaps in the spectra. There are two reasons why the computed eigenmode in Figure 6 (c) visually much better approximates the exact Fiedler vector compared to other cases in Figure 6. The first one is that the shape of the exact Fiedler eigenmode (blue line) is very pronounced and quite different from those of other eigenfunctions with small eigenvalues in Figure 4 left panel.

The second reason is more important and qualitative, but less evident, requiring additional information and explanations as follows. It is related to *condition numbers* of eigenvectors, which is primarily determined by a gap in the matrix spectrum.

The convergence speed of iterative approximation to an eigenvector, as well as eigenvector sensitivity with respect to perturbations in the matrix entries, e.g., due to noise, is mostly determined by a quantity, called the *condition number* of the eigenvector, defined for symmetric matrices as the ratio of the spread of the matrix spectrum to the gap in the eigenvalues. Since we are interested in the eigenvector, corresponding to the smallest eigenvalue, the gap is simply the difference between the smallest eigenvalue and the other nearest eigenvalue, where the trivial zero eigenvalue of the Laplacian can be excluded, since the corresponding trivial eigenvector, made of ones, can simply be ignored.

What happens in our example, as we see numerically, the largest eigenvalue remains basically the same for all variants, so we only need to check the gap. It turns out that the gap for the signed Laplacian is about 3 times smaller, for all tested values of the negative weight, compared the gap for the case of the zero weight, explaining why we see no improvement in Figures 6 (b) and (d), compared to (a). In contrast, introducing the negative weight in the original Laplacian tends to make the target smallest eigenvalue smaller, even negative, in our test for the discrete string, while not changing enough the other eigenvalues nearby. As a result, the gap with the negative weight -0.05 is 4 times larger compared to the baseline case of the zero weight.
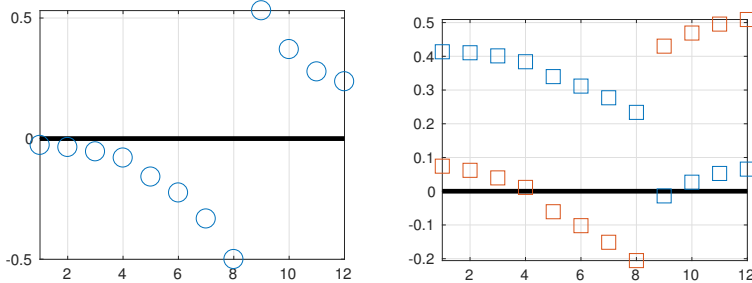
FIG. 7. *Laplacian eigenmodes, original (left) and signed (right), a "noisy" 12-mass string with a negative weight at one edge between vertices 8 and 9.*

We conclude that the eigenvector condition number for the signed Laplacian is about 3 time larger, while for the original Laplacian is 4 times smaller, depending on the negative weight -0.05, compared to the baseline eigenvector condition number for the Laplacian with the zero weight, suggesting that the signed Laplacian is numerically inferior for spectral clustering.

5.5. Signed Laplacian failure. We have observed that the approximate eigenvector of the signed Laplacian in Figure 6 (d) produces wrong clusters, although either the first (red) or the second (blue) exact eigenvector of the signed Laplacian in Figure 4 left panel would result in ideal clusters, cutting our linear graph at the correct location of the negative edge, between the 37 and 38 vertices. Now, we present a bit more sophisticated example, where neither the first nor the second exact eigenvectors of the signed Laplacian result in meaningful clusters, using the signs of the eigenvector components as suggested in [21]. In other words, the signed Laplacian fails.

We consider another standard linear mass-spring system with 12 masses and one repulsive spring, $w_{89} = w_{98} = -1/2$ between masses 8 and 9, but add to the graph adjacency an extra "noise" full random matrix with entrees uniformly distributed between 0 and 10^{-2}. It turns out that in this example the two smallest eigenvalues of the signed Laplacian form a cluster, making individual eigenvectors unstable with respect to the additive noise, leading to meaningless spectral clustering, if based on the signs on the components of any of the two eigenvectors.

The exact Laplacian eigenmodes, are shown in Figure 7, the original Fiedler (left panel) and both eigenvectors of the signed Laplacian (right panel). The signs of neither eigenvectors of the signed Laplacian give any meaningful clusters, while the Fiedler vector clearly suggests the perfect cut, as well as in our previous tests.

5.6. Signed Laplacian possible fix. A closer look at the two eigenvectors of the signed Laplacian in Figure 7 (right panel) reveals a visible aperture between the values of the components in every eigenvector at the actually correct location, between graph vertices 8 and 9, separating the graph partitions. This aperture, however, is not aligned with the sign change in any of the two eigenvectors, suggesting that partitioning, using the eigenvectors of the signed Laplacian, should be based on the location of the aperture, rather than the signs of the components of the eigenvector, in contrast to [21]. For example, the correct partitioning in this case can be determined via clustering of the one-dimensional point cloud, consisting of the components of the eigenvector, such as by K-means initiated with centroids at the local maximal cloud density points (see, e.g., [22] and references there), which in 1D is easy to estimate.

6. Benefits of negative weights for clustering. Clustering methods, which allows both the non-negative and negative entries in the graph adjacency matrix, are advantageous because situations where the pair of data points is disparate can be explicitly represented in the graph adjacency matrix. In the traditional approach, the entries in the graph adjacency matrix must be all non-negative, which does not admit negative data comparisons, available in many practical applications.

For example, a practically important case is where the data points are feature vectors of some underlining original data, or just vectors themselves. In such an example, the pair-wise comparing of data points can be determined based on a correlation or a covariance, instead of a distance, where the correlation and the covariance quantify a linear dependence between the vectors, such that the entry of the graph adjacency matrix is positive, negative, or zero, depending on whether the two vectors are positively correlated, negatively correlated, or uncorrelated, in contrast to [4], where negatively correlated data result in zero weights, while uncorrelated data give moderate positive weights.

Such a determination is advantageous because it directly uses the correlation or the covariance for the data comparing, even when the correlation or the covariance is negative for at least some pairs of the data points, thus potentially improving the quality of the clustering. In the traditional approach, the vectors would be typically compared using a kernel distance function leading to the graph adjacency matrix with all non-negative entries, enabling the conventional spectral clustering to proceed, but nullifying valid comparisons.

The repulsive phenomenon is beneficial for clustering, increasing the quality of distinguishing data points having negative similarities. In the vibration model of the mass-spring system, any eigenmode corresponding to a negative eigenvalue, contributes to increasing the amplitudes of vibrations of components when the interaction between the components is repulsive. The increasing of the amplitudes of vibrations of the components is an indication that the components are not likely to appear in the same cluster, which makes the determining of clusters simpler. This argument also suggests that the repulsive (negative) spectrum is more important compared to the attractive (positive) spectrum or even the zero spectrum, and thus should be treated preferentially, as we discuss below in § 7.

7. Dos and don'ts for negative weights. The graph adjacency matrix with both positive and negative entries can lead to the graph Laplacian having positive and negative eigenvalues. The repulsive spectrum of the negative eigenvalues of the graph Laplacian is impossible in the conventional approach, creating a need to adjust some conventional tools used in traditional spectral clustering.

For the purpose of determining the clustering, the repulsive spectrum is advantageous, since the left-most negative eigenvalue corresponds to the least stable eigenmode in the mass-spring wave equation (or the heat equation, cf. [25]) and, thus, is best suitable for determining the bisection partitioning using the signs of the components of the corresponding eigenvector.

In contrast to the conventional case of spectral clustering with all non-negative weights, the presence of the repulsive (negative) spectrum may apparently result in clustering, where every cluster does not have to be connected. The mathematical reason for the lost of the connectivity is that the corresponding classical proof is based on a maximum principle; see [7, 8]. In its turn, the maximum principle relies on negativity of all off-diagonal non-zero entries of the graph Laplacian matrix L, which is equivalent to non-negativity of all the entries of the graph adjacency matrix W.

Some results of the traditional spectral graph theory significantly use the assumption of the positivity of the non-zero graph edge weights, and may fail if the assumption is violated. For example, [17] bounds changes in the graph Laplacian spectra, by representing the graph Laplacian matrix as a diffusion operator on the vertexes of the graph, and then transforming the diffusion operator on the vertexes of the graph to a diffusion operator on the edges of the graph. The latter involves positive square roots of the graph edge weights, thus requiring weight positiveness.

Even if the diagonal degree matrix D is not degenerate, so that $P = D^{-1}W$ is correctly defined, some entries of P may be negative. In the conventional case of all non-negative entries of W, the matrix P is associated with a Markov random walk on the vertices of the graph, with the entries of P, also all non-negative giving the probabilities of the walk; e.g., see [27]. Since in our scenario the adjacency matrix W can have negative entries, so can P, failing the association with the random walk (unless some probabilities are allowed to be negative!). It is always possible to define $\bar{D}^{-1}W$, but it would not be row-stochastic, if negative entries are present.

8. Possible extensions for future work. We concentrate on the model of the system of masses connected with springs only because it directly leads to the standard definition of the graph Laplacian, giving us a simple way to justify our introduction of the negative weights. Similarly, we restrict the vibrations to be transversal, since then we can use the classical bisection partitioning definition based on the signs of the components of the Fiedler vector. It is important to realize that the negative weights can as well be introduced in other models, cf. [11], that intuitively perform clustering. Below we describe two more general models incorporating the negative weights.

The first model is based on vibration modes of a wave equation of a system of interacting quasi-particles subjected to vibrations. Each quasi-particle of the vibration model corresponds to one of the data points, and interaction coefficients of the vibration model are determined by pair-wise comparison of the data points, wherein the interacting is attractive and the interaction coefficient is positive if the data points in the pair are similar, or the interacting is absent and the interaction coefficient is zero when the data points in the pair are not comparable, or the interacting is repulsive and the interaction coefficient is negative when the data points in the pair are disparate, and where a strength of the interacting and an amplitude of the interaction coefficient represent a level of similarity or disparity.

The eigenmodes are defined as eigenvectors of an eigenvalue problem, resulted from the usual separation of the time and spacial variables. In low-frequency or unstable vibration modes, the quasi-particles are expected to move synchronically in the same direction if they are tightly connected by the attractive interactions, but in the opposite directions if the interactions are repulsive, or in the complementary (if available) directions if the interacting is absent.

Compared to the already considered transversal vibrations, where the masses can only move up or down, on the one hand, determining the clusters by analyzing the shapes of the vibrations is less straightforward than simply using the signs of the components, but, on the other hand, allows reliable detection of more than two clusters from a single eigenmode. For example, a quasi-particle representing an elementary volume of an elastic body in three-dimensional space has six degrees of freedom, allowing to reliably define up to twelve clusters from a single vibration mode. Already mentioned multiway spectral partitioning heuristic algorithms, such as in [5, 12, 31, 35], would have to be adopted to the new case, where a quasi-particle, associated with a graph vertex, has multiple degrees of freedom.

A second, alternative, model is based on the concentration-diffusion equation, for a system made of interacting quasi-particles, subjected to concentration or diffusion, where each quasi-particle of the concentration-diffusion model corresponds to a point in the data, and the model quasi-particle interaction conductivity coefficients are determined by pair-wise comparison of data points, wherein the interaction is diffusive and the interaction conductivity coefficient is positive if the data points in the pair are similar, or the interaction is absent and the interaction conductivity coefficient is zero, if the data points in the pair are not comparable, or the interaction is concentrative and the interaction conductivity coefficient is negative, if the data points in the pair are disparate. The strength of the interaction and the amplitude of the interaction coefficient represent the level of similarity or disparity.

As in the first model, the eigenvalue problem is obtained by the separation of the time and spacial variables in the time dependent diffusion equation. The clusters are defined by analyzing the eigenvectors corresponding to the left part of the spectrum, describing unstable or slowest modes. Quasi-particles that concentrate together form a cluster by definition.

A forward-and-backward diffusion in [10, 34] provides a different interpretation of a similar diffusion equation, but where the negative sign in the conductivity coefficient is moved to the time derivative, thus making the time to go backward. This model requires the time to go forward on the graph edges with with the positive weights and backward on the edges with the negative weights.

Conclusions. Spectral clustering has been successful in many applications, ranging from traditional resource allocation, image segmentation, and information retrieval, to more recent bio- and material-informatics, providing good results at reasonable costs. Improvements of cluster quality and algorithm performance are important, e.g., for big data or real-time clustering. We discuss introducing negative weights in the graph adjacency matrix for incorporating disparities in data via the powerful technique of spectral clustering that traditionally handles data with similarities.

The direct incorporating of the disparities in the data into spectral clustering is expected to be of significance and have impact in any application domain, where the data disparities naturally appear, e.g., if the data comparison involves the correlation or the covariance. If the data or data features are represented by elements of a vector space equipped with a vector scalar product, it can be used for determining the pairwise comparison function, having both negative and non-negative values.

Our arguments aim at convincing the reader that the main framework of the traditional spectral clustering, with only non-negative weights, remains largely intact when negative weights are present. For example, the eigenvectors corresponding to the algebraically smallest eigenvalues (that can be negative) of the graph Laplacian allow defining the clusters, at similar computational costs, but higher quality, compared to a naive substitution dissimilarities for disparities, to preserve the weights non-negative, as required in the traditional technique, or compared to the signed Laplacian.

For signed graphs, we argue that the standard Laplacian may outperform the signed Laplacian for spectral clustering, and is better justified, using both the mass-spring system model with repulsive springs and the spectral clustering introduction via relaxation of the combinatorial minimization. The repulsive phenomenon is beneficial for clustering, increasing the quality of distinguishing data points having negative similarities, as well as increasing the relevant gap in the spectrum of the original graph Laplacian, making the Fiedler vector easier to compute and more robust to noise, compared to the eigenvectors of the signed Laplacian.

References.

[1] H. Anzt, S. Tomov, and J. Dongarra, *On the performance and energy efficiency of sparse linear algebra on gpus*, The International Journal of High Performance Computing Applications, 0 (0), p. 1094342016672081, https://doi.org/10.1177/1094342016672081.

[2] N. Bansal, A. Blum, and S. Chawla, *Correlation clustering*, Machine Learning, 56 (2004), pp. 89–113, http://dx.doi.org/10.1023/B:MACH.0000033116.57574.95. doi:10.1023/B:MACH.0000033116.57574.95.

[3] J. C. Bronski and L. DeVille, *Spectral theory for dynamics on graphs containing attractive and repulsive interactions*, SIAM Journal on Applied Mathematics, 74 (2014), pp. 83–105, https://doi.org/10.1137/130913973, http://dx.doi.org/10.1137/130913973. doi:10.1137/130913973.

[4] W. Casaca, A. Paiva, and L. Nonato, *Spectral segmentation using cartoon-texture decomposition and inner product-based metric*, in Graphics, Patterns and Images (Sibgrapi), 2011 24th SIBGRAPI Conference on, Aug 2011, pp. 266–273, https://doi.org/10.1109/SIBGRAPI.2011.34. doi:10.1109/SIBGRAPI.2011.34.

[5] P. K. Chan, M. D. F. Schlag, and J. Y. Zien, *Spectral k-way ratio-cut partitioning and clustering*, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 13 (1994), pp. 1088–1096, https://doi.org/10.1109/43.310898.

[6] M. Cucuringu, I. Koutis, S. Chawla, G. Miller, R. Peng, and C. Qatar, *Simple and scalable constrained clustering: A generalized spectral method*, in Proceedings of the 19th International Conference on Artificial Intelligence and Statistics, 2016, pp. 445–454, http://www.jmlr.org/proceedings/papers/v51/cucuringu16.pdf.

[7] M. Fiedler, *Algebraic connectivity of graphs*, Czechoslovak mathematical journal, 23 (1973), pp. 298–305. http://dml.cz/handle/10338.dmlcz/101168.

[8] M. Fiedler, *A property of eigenvectors of nonnegative symmetric matrices and its application to graph theory*, Czechoslovak Mathematical Journal, 25 (1975), pp. 619–633. http://hdl.handle.net/10338.dmlcz/101357.

[9] J. Gallier, *Spectral Theory of Unsigned and Signed Graphs. Applications to Graph Clustering: a Survey*, ArXiv e-prints, (2016), https://arxiv.org/abs/1601.04692. http://adsabs.harvard.edu/abs/2016arXiv160104692G.

[10] G. Gilboa, N. Sochen, and Y. Y. Zeevi, *Forward-and-backward diffusion processes for adaptive image enhancement and denoising*, IEEE Transactions on Image Processing, 11 (2002), pp. 689–703. doi:10.1109/TIP.2002.800883.

[11] J. Ham, D. D. Lee, S. Mika, and B. Schölkopf, *A kernel view of the dimensionality reduction of manifolds*, in Proceedings of the Twenty-first International Conference on Machine Learning, ICML '04, New York, NY, USA, 2004, ACM, pp. 47–, http://doi.acm.org/10.1145/1015330.1015417. doi:10.1145/1015330.1015417.

[12] R. Jothi, S. K. Mohanty, and A. Ojha, *On the Impact of Post-clustering Phase inMulti-way Spectral Partitioning*, Springer International Publishing, Cham, 2015, pp. 161–169, https://doi.org/10.1007/978-3-319-26832-3_16.

[13] A. Knyazev, *Toward the optimal preconditioned eigensolver: locally optimal block preconditioned conjugate gradient method*, SIAM J. Scientific Computing, 23 (2001), pp. 517–541. doi:10.1561/0600000020.

[14] A. Knyazev, *Edge-enhancing filters with negative weights*, in 2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP), Dec 2015, pp. 260–264. doi:10.1109/GlobalSIP.2015.7418197.

[15] A. KNYAZEV AND A. MALYSHEV, *Conjugate gradient acceleration of non-linear smoothing filters*, in 2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP), Dec 2015, pp. 245–249. doi:10.1109/GlobalSIP.2015.7418194.

[16] A. V. KNYAZEV, *Modern preconditioned eigensolvers for spectral image segmentation and graph bisection*, in Proceedings of the workshop Clustering Large Data Sets; Third IEEE International Conference on Data Mining (ICDM 2003), Boley, Dhillon, Ghosh, and Kogan, eds., Melbourne, Florida, 2003, IEEE Computer Society, pp. 59–62. http://math.ucdenver.edu/~aknyazev/research/conf/ICDM03.pdf.

[17] A. V. KNYAZEV AND M. E. ARGENTATI, *Majorization for changes in angles between subspaces, Ritz values, and graph Laplacian spectra*, SIAM Journal on Matrix Analysis and Applications, 29 (2007), pp. 15–32, https://doi.org/10.1137/060649070. doi:10.1137/060649070.

[18] A. V. KNYAZEV, M. E. ARGENTATI, I. LASHUK, AND E. E. OVTCHINNIKOV, *Block Locally Optimal Preconditioned Eigenvalue Xolvers (BLOPEX) in Hypre and PETSc.*, SIAM J. Scientific Computing, 29 (2007), pp. 2224–2239. doi:10.1137/060661624.

[19] R. KOLLURI, J. R. SHEWCHUK, AND J. F. O'BRIEN, *Spectral surface reconstruction from noisy point clouds*, in Proceedings of the 2004 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing, SGP '04, New York, NY, USA, 2004, pp. 11–21, http://doi.acm.org/10.1145/1057432.1057434. doi:10.1145/1057432.1057434.

[20] J. KUNEGIS, A. LOMMATZSCH, AND C. BAUCKHAGE, *The Slashdot Zoo: Mining a social network with negative edges*, in Proceedings of the 18th International Conference on World Wide Web, WWW '09, New York, NY, USA, 2009, ACM, pp. 741–750, http://doi.acm.org/10.1145/1526709.1526809. doi:10.1145/1526709.1526809.

[21] J. KUNEGIS, S. SCHMIDT, A. LOMMATZSCH, J. LERNER, E. W. D. LUCA, AND S. ALBAYRAK, *Spectral analysis of signed graphs for clustering, prediction and visualization*, in Proceedings of the 2010 SIAM International Conference on Data Mining, SIAM, 2010, pp. 559–570, http://epubs.siam.org/doi/abs/10.1137/1.9781611972801.49. doi:10.1137/1.9781611972801.49.

[22] F. LI, H. QIAO, AND B. ZHANG, *Effective deterministic initialization for k-means-like methods via local density peaks searching*, CoRR, abs/1611.06777 (2016), http://arxiv.org/abs/1611.06777.

[23] J. LIU AND J. HAN, *Spectral clustering*, in Data Clustering: Algorithms and Applications, C. Aggarwal and C. Reddy, eds., Chapman and Hall/CRC Press, 2013. http://www.crcnetbase.com/doi/pdf/10.1201/b15410-9.

[24] U. LUXBURG, *A tutorial on spectral clustering*, Statistics and Computing, 17 (2007), pp. 395–416, https://doi.org/10.1007/s11222-007-9033-z. doi:10.1007/s11222-007-9033-z.

[25] J. MA, W. HUANG, S. SEGARRA, AND A. RIBEIRO, *Diffusion filtering of graph signals and its use in recommendation systems*, in 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), March 2016, pp. 4563–4567. doi:10.1109/ICASSP.2016.7472541.

[26] J. MCQUEEN, M. MEILA, J. VANDERPLAS, AND Z. ZHANG, *megaman: Manifold learning with millions of points*, CoRR, abs/1603.02763 (2016), http://arxiv.org/abs/1603.02763.

[27] M. MEILA AND J. SHI, *Learning segmentation by random walks*, in Advances

in Neural Information Processing Systems 13, T. K. Leen, T. G. Dietterich, and V. Tresp, eds., MIT Press, 2001, pp. 873–879. http://papers.nips.cc/paper/1830-learning-segmentation-by-random-walks.pdf.

[28] A. Y. Ng, M. I. Jordan, and Y. Weiss, *On spectral clustering: Analysis and an algorithm*, in Advances in Neural Information Processing Systems 14, T. G. Dietterich, S. Becker, and Z. Ghahramani, eds., MIT Press, 2002, pp. 849–856. http://papers.nips.cc/paper/2092-on-spectral-clustering-analysis-and-an-algorithm.pdf.

[29] J. Park, M. Jeon, and W. Pedrycz, *Spectral clustering with physical intuition on spring-mass dynamics*, J. Franklin Institute, 351 (2014), pp. 3245 – 3268, http://www.sciencedirect.com/science/article/pii/S0016003214000532. doi:10.1016/j.jfranklin.2014.02.017.

[30] A. Pothen, H. D. Simon, and K.-P. Liou, *Partitioning sparse matrices with eigenvectors of graphs*, SIAM Journal on Matrix Analysis and Applications, 11 (1990), pp. 430–452, https://doi.org/10.1137/0611030, http://dx.doi.org/10.1137/0611030.

[31] M. A. Riolo and M. E. J. Newman, *First-principles multiway spectral partitioning of graphs*, Journal of Complex Networks, 2 (2014), p. 121, https://doi.org/10.1093/comnet/cnt021.

[32] L. Rossi, A. Torsello, and E. R. Hancock, *Unfolding kernel embeddings of graphs: Enhancing class separation through manifold learning*, Pattern Recognition, (2015), pp. –, https://doi.org/http://dx.doi.org/10.1016/j.patcog.2015.03.018, http://www.sciencedirect.com/science/article/pii/S0031320315001193. doi:10.1016/j.patcog.2015.03.018.

[33] J. Shi and J. Malik, *Normalized cuts and image segmentation*, IEEE Trans. Pattern Anal. Mach. Intell., 22 (2000), pp. 888–905, http://dx.doi.org/10.1109/34.868688. doi:10.1109/34.868688.

[34] L. Tang and Z. Fang, *Edge and contrast preserving in total variation image denoising*, EURASIP Journal on Advances in Signal Processing, 2016 (2016), pp. 1–21, https://doi.org/10.1186/s13634-016-0315-5, http://dx.doi.org/10.1186/s13634-016-0315-5. doi:10.1186/s13634-016-0315-5.

[35] J. Voss, M. Belkin, and L. Rademacher, *The hidden convexity of spectral clustering*, 2016, http://www.aaai.org/ocs/index.php/AAAI/AAAI16/paper/view/12307.

[36] M. M. Wolf, A. M. Klinvex, and D. M. Dunlavy, *Advantages to modeling relational data using hypergraphs versus graphs*, in 2016 IEEE High Performance Extreme Computing Conference (HPEC), Sept 2016, pp. 1–7, https://doi.org/10.1109/HPEC.2016.7761624.

[37] L. Wu, X. Wu, A. Lu, and Y. Li, *On spectral analysis of signed and dispute graphs*, in 2014 IEEE International Conference on Data Mining, Dec 2014, pp. 1049–1054. doi:10.1109/ICDM.2014.113.