Combinatorial Fiedler theory and graph partition

Enide Andrade [a], Geir Dahl [b,*]

[a] *Center for Research and Development in Mathematics and Applications, Department of Mathematics, University of Aveiro, Portugal*
[b] *Department of Mathematics, University of Oslo, Norway*

A R T I C L E   I N F O

A B S T R A C T

Partition problems in graphs are extremely important in applications, as shown in the Data Science and Machine Learning literature. One approach is spectral partitioning based on a Fiedler vector, i.e., an eigenvector corresponding to the second smallest eigenvalue $a(G)$ of the Laplacian matrix L_G of the graph G. This problem corresponds to the minimization of a quadratic form associated with L_G, under certain constraints involving the ℓ_2-norm. We introduce and investigate a similar problem, but using the ℓ_1-norm to measure distances. This leads to a new parameter $b(G)$ as the optimal value. We show that a well-known cut problem arises in this approach, namely the sparsest cut problem. We prove connectivity results and different bounds on this new parameter, relate to Fiedler theory and show explicit expressions for $b(G)$ for trees. We also comment on an ℓ_∞-norm version of the problem.

1. Introduction

Often real world networks contain clusters, that is, groups of points each with a large number of neighbors among them and not many connections to the outside. In Data

* Corresponding author.
  E-mail addresses: enide@ua.pt (E. Andrade), geird@math.uio.no (G. Dahl).

Science and Machine Learning the task of *clustering* is very important. Given a set of data points (in some space) and their common properties measured in terms of distances, the clustering problem consists in finding subsets of these data points that are "similar". If the points to be clustered are vertices in a graph, and the edges connecting these vertices are the only information available, then the problem is called *graph clustering* [20,25]. There are many methods for graph clustering, and one popular such method is *spectral clustering*. The basis is then spectral bisection where a Fiedler vector [15] is used for partitioning a graph into two connected subgraphs based on the signs of the components of the vector. This splitting may be repeated for each of the parts and thereby obtain a desired partition.

Consider an unweighted (undirected) simple graph $G = (V, E)$ and let $n = |V|$. Throughout the paper we assume that G is connected. Recall that the *Laplacian matrix* L_G is the $n \times n$ matrix $L_G = D_G - A_G$ where D_G is the diagonal matrix with the vertex degrees on the diagonal, and A_G is the adjacency matrix of G. The matrix L_G is positive semidefinite and therefore it has only real, nonnegative eigenvalues. The algebraic connectivity $a(G)$ is the second smallest eigenvalue of L_G, and it is known as a connectivity measure in the graph [14,23]. In particular, G is connected if and only if $a(G) > 0$. The matrix L_G is singular, 0 is the smallest eigenvalue and a corresponding eigenvector is the all ones vector e. Therefore, by the Courant-Fischer theorem [16],

$$a(G) = \min\{x^T L_G\, x : e^T x = 0,\ \|x\|_2 = 1\}.$$

Here $\|x\|_2 = (\sum_i x_i^2)^{1/2}$ is the (Euclidean) ℓ_2-norm of $x = (x_1, x_2, \ldots, x_n) \in \mathbb{R}^n$. By using a standard factorization $L_G = B^T B$ where B is the oriented edge-vertex incidence matrix of G, we have the alternative expression

$$a(G) = \min\{ \sum_{uv \in E} (x_u - x_v)^2 : \sum_{v \in V} x_v = 0,\ \|x\|_2 = 1\}. \tag{1}$$

Thus, an eigenvector corresponding to the eigenvalue $a(G)$, usually called a *Fiedler vector*, can be seen as assigning values to the vertices to obtain an optimal "smoothing" along edges, i.e., small difference between end points of edges, under the two normalization constraints, see [26]. These constraints assure that we avoid a constant solution $x = \lambda e$ for some λ and, also, the norm constraint avoids scaling (to get similar solutions) and the zero vector. We shall therefore call the minimization problem in (1) the ℓ_2-*graph smoothing* problem. Here ℓ_2 refers to the fact that both the (objective) function $\sum_{uv \in E}(x_u - x_v)^2$ and the norm constraint involve the ℓ_2-norm.

The motivating question of our study is:

- What happens if we modify the optimization problem (1) by changing the norm involved to the ℓ_1-norm (the sum norm)?

Problem:	ℓ_2-graph smoothing	ℓ_1-graph smoothing
Norm:	ℓ_2-norm	ℓ_1-norm
Minimize:	$\sum_{uv \in E}(x_u - x_v)^2$	$\sum_{uv \in E}|x_u - x_v|$
Constraints:	$\|x\|_2 = 1,\ x \perp e$	$\|x\|_1 = 1,\ x \perp e$
Solution:	Fiedler vector, partition	ℓ_1-Fiedler vector, sparsest cut
Optimal value:	algebraic connectivity $a(G)$	$b(G)$
Property:	connected parts	connected parts
Approach:	spectral	combinatorial

Fig. 1. Graph smoothing and partition problems.

The ℓ_1-norm of $x = (x_1, x_2, \ldots, x_n) \in \mathbb{R}^n$ is defined by $\|x\|_1 = \sum_i |x_i|$. In fact, often in mathematics different norms may be used in the study of some (approximation) problem. A well-known such example is the linear approximation problem $\min_{x \in \mathbb{R}^n} \|Cx - b\|$ where C is an $m \times n$ (real) matrix, $b \in \mathbb{R}^m$ and $\|\cdot\|$ is some vector norm. Then, for the ℓ_2-norm, we obtain the least squares problem, and for the ℓ_1-norm and ℓ_∞-norm one may use linear programming to solve the problem. It is important to understand the properties of solutions, and how they depend on the choice of norm.

Therefore, our main goal is to consider a new graph smoothing problem which we call ℓ_1-*graph smoothing*. It is similar to ℓ_2-graph smoothing except that we change the norm from the ℓ_2-norm to the ℓ_1-norm. Let $G = (V, E)$ be a given connected graph with at least one edge. The ℓ_1-*graph smoothing* problem is the following optimization problem

$$b(G) = \min\{ \sum_{uv \in E} |x_u - x_v| : \sum_{v \in V} x_v = 0,\ \|x\|_1 = 1 \}. \tag{2}$$

Clearly the minimum here is attained by some x as the constraints define a compact set and the function to be minimized is continuous. Note that the constraint set is not a convex set. An optimal solution x in (2) will be called an ℓ_1-*Fiedler vector*. Then x satisfies $\sum_v x_v = 0$, $\|x\|_1 = 1$ and $\sum_{uv \in E} |x_u - x_v| = b(G)$.

A main contribution of our paper is indicated in Fig. 1 (where $u \perp v$ for vectors u and v in $\mathbb{R}^n$ means $u^T v = 0$). The two graph smoothing problems are indicated in the two last columns of the figure. One sees how the two problems are quite similar. Based on several intermediate results we establish that optimal solutions in the new problem correspond to so-called sparsest cuts. This also means that strong connections, with bounds, between the two optimal values $a(G)$ and $b(G)$ may be found. Moreover, there are important consequences in terms of computational complexity. A main contribution of this paper is to show that *there is a very natural optimization approach that is underlying sparsest cuts*. Thus, the problem (2) can be handled using a *combinatorial approach*. For related combinatorial approaches to Perron values of trees, see [4,5]. We will discuss the computational complexity for these graph smoothing problems, although this is not done in any detail. However, some remarks are given on the complexity of approximation problems related to $b(G)$, and we believe more theoretical work can be done here. The approaches discussed in [9] for approximate graph coloring may be of interest in this connection. We remark that a slightly related problem to (2) is the minimum p-sum

problem [17] where one minimizes the variation over the edges measured in p-norm taken over all possible labelings $1, 2, \ldots, n$ of the vertices.

The remaining part of this Introduction is devoted to some more results from spectral partitioning and relevant spectral graph theory. Finally, an overview of the next sections is given.

In [29] one studies the maximal error in spectral bisection where the two parts in the partition have the same size. In [18] the authors investigate graphs having Fiedler vectors with unbalanced sign patterns such that a partition can result in two connected subgraphs that are distinctly different in size. They also characterize graphs with a Fiedler vector having exactly one negative component. Motivated by these results we recall some facts concerning spectral partitioning and Cheeger bounds. For a much deeper discussion of these areas, we refer to the lectures notes [27] and [28], and the references found there. Consider again a graph $G = (V, E)$ and let $n = |V|$. Let $S \subseteq V$ be a nonempty vertex set, where $S \neq V$. The *cut* induced by S, denoted by $\delta(S)$, is the set of edges uv where $u \in S$ and $v \notin S$. Cuts are important objects in graph theory, combinatorial optimization as well as in applications. Define the *relative cut size*

$$\xi(S) = \frac{|\delta(S)|}{|S|}, \tag{3}$$

which is the size of the cut relative to the size of the vertex set S. This is an important notion in this paper. The *isoperimetric number* of G, also called *Cheeger's constant*, is the parameter

$$i(G) = \min_S \xi(S),$$

where the minimum is taken over all nonempty subsets S of V with $|S| \leq \lfloor n/2 \rfloor$.

A basic treatment of the isoperimetric number and its properties may be found in [22]. A related notion [21] is the *edge density* of a cut, defined as follows,

$$\rho(S) = \frac{|\delta(S)|}{|S||V \setminus S|}.$$

This concept represents the density of the edges in G between the set S and its complement, compared to the number of edges in a complete bipartite graph (with vertex set S and its complement). This parameter $\rho(S)$ is also called the *sparsity* of the cut. A *sparsest cut* is a cut which minimizes $\rho(S)$. For more literature related to these concepts, see [12].

A small calculation shows the following relation between edge density and relative cut sizes for each subset S (with $\emptyset \subset S \subset V$)

$$\xi(S) + \xi(V \setminus S) = n\rho(S). \tag{4}$$

Below we give some inequalities relating edge density and Laplacian eigenvalues, in particular the algebraic connectivity.

Theorem 1.1 *([21]). Let G be a graph of order n. For any nonempty subset S of vertices of G, $S \neq V$, the edge density is uniformly bounded below and above by*

$$\frac{a(G)}{n} \leq \rho(S) \leq \frac{\lambda_1}{n},$$

where λ_1 is the largest eigenvalue of $L(G)$.

In [13] one characterizes the graphs for which $a(G) = \rho(S)$, for some subset S of vertices.

There is another upper bound on the minimal density of cuts in terms of $a(G)$.

Theorem 1.2 *([21]). Let $G = (V, E)$ be a graph of order n with at least two edges. Then*

$$\min\{\rho(S) : S \subset V, S \neq \emptyset\} \leq \frac{2}{n}\sqrt{a(G)[2d_{\max}(G) - a(G)]},$$

where $d_{\max}(G)$ is the maximal vertex degree in G.

This upper bound is a strong discrete version of the well-known *Cheeger's inequality* from differential geometry [10], bounding the first eigenvalue of a Riemannian manifold. It appeared in [1,2] and later, as an improved edge version in [22].

The remaining paper is organized as follows. In Section 2 we study the set of feasible solutions of the ℓ_1-graph smoothing problem and present a rewriting of the problem. The main results are then presented in Section 3, where it is shown that optimal solutions correspond to sparsest cuts with a connectivity property. The computational complexity is also settled. In Section 4 a comparison of $b(G)$ and other parameters is made. Section 5 is devoted to examples and specific classes of graphs where explicit expressions for $b(G)$ are found. Moreover, a computational example is shown. In the final section we briefly consider a graph smoothing problem based on the ℓ_∞-norm.

Notation: Vectors in $\mathbb{R}^n$ are considered as column vectors and identified with the real n-tuples. The i'th component of a vector $x \in \mathbb{R}^n$ is usually denoted by x_i ($i \leq n$). A zero matrix, or vector, is denoted by O, and an all ones vector is denoted by e (the dimension should be clear from the context). For a real number c define $c^+ = \max\{c, 0\}$ and $c^- = \max\{-c, 0\}$. Then $c = c^+ - c^-$ and $|c| = c^+ + c^-$. Let $x = (x_1, x_2, \ldots, x_n) \in \mathbb{R}^n$. Define $x^+ = (x_1^+, x_2^+, \ldots, x_n^+) \in \mathbb{R}^n$ and $x^- = (x_1^-, x_2^-, \ldots, x_n^-) \in \mathbb{R}^n$. Thus, $x = x^+ - x^-$. In a graph we let $P = v_1, v_2, \ldots, v_k$ denote the path with vertices $v_1, v_2, \ldots, v_k$ in that order, so $v_i v_{i+1}$ is an edge ($i < k$). Moreover, P_n (resp. S_n) is a path (resp. the star) with n vertices, and K_n represents the complete graph with n vertices.

2. The ℓ_1-graph smoothing problem

As mentioned, our main goal is to introduce and study the ℓ_1-graph smoothing problem. In this section we shall rewrite the problem into a convenient form.

Throughout, we let the vertex set of the graph G be $V = \{v_1, v_2, \ldots, v_n\}$, and we identify a function $x \in \mathbb{R}^V$ with the vector $x = (x_1, x_2, \ldots, x_n)$ where $x_j = x(v_j)$ for each $j \leq n$. Define F_1 as the *feasible set* in (2), i.e.,

$$F_1 = \{x \in \mathbb{R}^n : \sum_j x_j = 0, \ \|x\|_1 = 1\}.$$

We also define the "smoothing function" $f_1(x) = \sum_{uv \in E} |x_u - x_v|$. Therefore, the ℓ_1-graph smoothing problem is to minimize $f_1(x)$ subject to $x \in F_1$.

Example 1. Consider $P_4 = v_1, v_2, v_3, v_4$, the path with four vertices. Let $x^1 = (-1/2, 0, 0, 1/2)$, so the only nonzeros are in the end vertices. Then $x^1 \in F_1$ and $f_1(x^1) = 1/2 + 0 + 1/2 = 1$. Next, consider $x^2 = (-1/4, -1/4, 1/4, 1/4)$. Then $x^2 \in F_1$ and $f_1(x^2) = 0 + 1/2 + 0 = 1/2$. So, x^2 is better than x^1 and $b(P_4) \leq 1/2$. But is x^2 optimal? The answer is yes, as will follow from later results. We remark that the algebraic connectivity of P_4 is given by $a(P_4) = 0.5858$.

An alternative description of the feasible set F_1 is given next.

Lemma 2.1.

$$F_1 = \{x \in \mathbb{R}^n : \sum_{j:x_j \geq 0} x_j = 1/2, \ \sum_{j:x_j \leq 0} x_j = -1/2\}. \tag{5}$$

Proof. Let $x \in F_1$. Then

$$1 = \|x\|_1 = \sum_{j:x_j \geq 0} x_j - \sum_{j:x_j \leq 0} x_j,$$

and adding this equation to $\sum_j x_j = 0$ gives $2 \sum_{j:x_j \geq 0} x_j = 1$, i.e., $\sum_{j:x_j \geq 0} x_j = 1/2$. This implies $\sum_{j:x_j \leq 0} x_j = -1/2$ as $\sum_j x_j = 0$. Conversely, if $\sum_{j:x_j \geq 0} x_j = 1/2$ and $\sum_{j:x_j \leq 0} x_j = -1/2$, then $\sum_j x_j = 0$ and $\|x\|_1 = 1$, by the equation above. $\square$

A vector $x \in F_1$ will be called a *feasible* solution of (2). Thus, by Lemma 2.1, a feasible solution partitions the vertices into three subsets depending on the sign of each x_j, ± 1 or 0, and the sum of the components of x in the positive and negative part is the same in absolute value, $\sum_{j:x_j > 0} x_j = |\sum_{j:x_j < 0} x_j| = 1/2$.

We next rewrite problem (2). Define

$$\beta(G) = \min\{\sum_{ij \in E} |x_i^1 - x_i^2 - x_j^1 + x_j^2| : \sum_{j=1}^{n} x_j^1 = 1/2, \sum_{j=1}^{n} x_j^2 = 1/2, (x^1)^T x^2 = 0, x^1, x^2 \geq O\}$$

(6)

where $x^1 = (x_1^1, x_2^1, \ldots, x_n^1)$ and $x^2 = (x_1^2, x_2^2, \ldots, x_n^2)$ are vectors in $\mathbb{R}^n$. Note that the constraints assure that for each i at least one of the two variables x_i^1 and x_i^2 is zero. The next result connects the two optimization problems (2) and (6).

Lemma 2.2. *The following holds:*
  (i) If x is optimal in (2), then $x^1 = x^+$, $x^2 = x^-$ is optimal in (6).
  (ii) If x^1, x^2 are optimal in (6), then $x = x^1 - x^2$ is optimal in (2).
  (iii) $b(G) = \beta(G)$.

Proof. Properties (i) and (ii) follow by replacing each variable x_j by $x_j = x_j^1 - x_j^2$ where x_j^1 and x_j^2 are two nonnegative variables. In this construction x^1 and x^2 are only unique up to a positive additive constant in each term, but the orthogonality constraint $(x^1)^T x^2$ assures uniqueness and that $x^1 = x^+$ and $x^2 = x^-$. Thus (6) is a reformulation of (2). This implies that the optimal values coincide, so (iii) holds. □

Later we prove that it is *NP*-hard to compute $b(G)$ and a corresponding ℓ_1-Fiedler vector. Still, the previous lemma means that computing $b(G)$, and the corresponding optimal x, may be done by solving (6). This is a problem of minimizing a piecewise linear convex function subject to linear constraints and a "complementarity constraint" saying that $x_i^1 x_i^2 = 0$ for each i. This problem can be written as a linear programming problem with certain linear complementarity constraints corresponding to the orthogonality $x^1 \perp x^2$, as explained next. Consider the following optimization problem with variables x_j^1, x_j^2 ($j \leq n$) and y_{ij} for $ij \in E$.

$$
\begin{aligned}
\text{minimize} \qquad & \sum_{ij \in E} y_{ij} \\
x_i^1 - x_i^2 - x_j^1 + x_j^2 & \leq y_{ij} \qquad (ij \in E), \\
-x_i^1 + x_i^2 + x_j^1 - x_j^2 & \leq y_{ij} \qquad (ij \in E), \\
\sum_{j=1}^{n} x_j^1 & = 1/2, \\
\sum_{j=1}^{n} x_j^2 & = 1/2, \\
x_j^1 x_j^2 & = 0 \qquad (j \leq n), \\
x^1 \geq O, \ x^2 \geq O, \ y & \in \mathbb{R}^E.
\end{aligned}
$$

(7)

In fact, the first two constraints are equivalent to $|x_i^1 - x_i^2 - x_j^1 + x_j^2| \leq y_{ij}$ ($ij \in E$) and due to the minimization equality must hold here for every $ij \in E$. As mentioned, the ℓ_1-graph smoothing problem is *NP-hard*, but several general integer programming based algorithms have been developed that may be used to give approximate solutions

of (7). A basic reference on the linear complementarity problem is [11]. We leave it as an interesting idea for further research to use this formulation in order to find approximate solutions of the ℓ_1-graph smoothing problem. In the final section of this paper we also use a related linear programming approach to a graph smoothing problem based on the ℓ_∞-norm.

3. ℓ_1-graph smoothing and sparsest cuts

In this section we investigate the ℓ_1-graph smoothing problem closer and establish strong properties of the optimal solutions. This leads to a connection to sparsest cuts. Recall that we assume that the graph G is connected. The proof of the next result uses a construction based on the relative cut size $\xi(S)$ defined in (3).

Theorem 3.1. *Let x be an ℓ_1-Fiedler vector with the maximum number of zeros. Then the subgraph induced by $\{v \in V : x_v > 0\}$ is connected and the subgraph induced by $\{v \in V : x_v < 0\}$ is connected.*

Proof. We shall first prove that the subgraph induced by $V^+ := \{v \in V : x_v > 0\}$ is connected. The proof is by contradiction, so assume the subgraph induced by V^+ is *not* connected. Then there must exist disjoint subsets S_1 and S_2 of V^+ such that

(i) no edge joins S_1 and S_2,
(ii) $x_v > 0$ $(v \in S_1 \cup S_2)$,
(iii) if $vw \in \delta(S_1)$ with $v \in S_1$ (so $w \notin S_1$) then $x_w \leq 0$.

We discuss different cases.

Case 1: $\xi(S_1) > \xi(S_2)$. Let ϵ be a "suitably small" positive number; in fact, $\epsilon < \min\{x_v : v \in V^+\}$ works. Define $y \in \mathbb{R}^V$ based on x as follows:

$$
y_v = \begin{cases}
x_v - \epsilon & (v \in S_1), \\
x_v + (|S_1|/|S_2|)\epsilon & (v \in S_2), \\
x_v & (\text{otherwise}).
\end{cases}
$$

Then

$$
\sum_{v:y_v \geq 0} y_v - \sum_{v:x_v \geq 0} x_v = \sum_{v \in S_1}(-\epsilon) + \sum_{v \in S_2}(|S_1|/|S_2|)\epsilon = -|S_1|\epsilon + |S_1|\epsilon = 0.
$$

Therefore,

$$
\sum_{v:y_v \geq 0} y_v = \sum_{v:x_v \geq 0} x_v = 1/2.
$$

Moreover,

$$\sum_{v:y_v \leq 0} y_v = \sum_{v:x_v \leq 0} x_v = -1/2;$$

recall that $y_v = x_v$, for all $v \notin (S_1 \cup S_2)$. This proves that $y \in F_1$.

Next, for each edge $vw \in \delta(S_1)$ with $v \in S_1$, $|y_v - y_w| = (x_v - x_w) - \epsilon$ as $x_v > 0 \geq x_w$. Similarly, for each edge $vw \in \delta(S_2)$ with $v \in S_2$, $|y_v - y_w| = (x_v - x_w) + (|S_1|/|S_2|)\epsilon$. Therefore

$$
\begin{aligned}
f_1(y) - f_1(x) &= -\epsilon|\delta(S_1)| + (|S_1|/|S_2|)\epsilon|\delta(S_2)| \\
&= -\epsilon\big(|\delta(S_1)| - (|S_1|/|S_2|)|\delta(S_2)|\big) \\
&< 0
\end{aligned}
\tag{8}
$$

as $\xi(S_1) > \xi(S_2)$ means $|\delta(S_1)|/|S_1| > |\delta(S_2)|/|S_2|$ so $|\delta(S_1)| - (|S_1|/|S_2|)|\delta(S_2)| > 0$. This proves that $b(G) \leq f_1(y) < f_1(x)$, contradicting that x is an ℓ_1-Fiedler vector.

Case 2: $\xi(S_1) < \xi(S_2)$. By symmetry of S_1 and S_2 this can be treated by similar arguments and a contradiction is derived.

Case 3: $\xi(S_1) = \xi(S_2)$. We use the same construction of the vector y as in Case 1, but we let $\epsilon = \min\{x_v : v \in S_1\}$. Then $y \in F_1$, and from (8) we see that $f_1(y) = f_1(x)$, so y is also an ℓ_1-Fiedler vector. However, y has at least one more zero than x, and this contradicts the choice of x (initially in the proof).

Thus, in each case we obtained a contradiction, which proves that the subgraph induced by $V^+ := \{v \in V : x_v > 0\}$ is connected. The proof that the subgraph induced by $\{v \in V : x_v < 0\}$ is connected is completely similar, by choosing $\epsilon < \min\{|x_v|, v \in V^-\}$ in Case 1 and $\epsilon = \min\{|v_v|, v \in S_1\}$ in Case 3. $\square$

Next we give a main result which shows an explicit formula for $b(G)$ which is of a combinatorial nature. This will give a strong connection to the notions presented in the Introduction.

We say that the pair (S_1, S_2) is a *quasi-bipartition* of V if

$$S_1, S_2 \subset V, \ S_1 \cap S_2 = \emptyset, \text{ and } S_1, S_2 \neq \emptyset.$$

For a quasi-bipartition (S_1, S_2) let $x = x^{(S_1, S_2)} = (x_v : v \in V) \in \mathbb{R}^n$ be the vector with

$$
x_v = \begin{cases}
\frac{1}{2|S_1|} & (v \in S_1), \\
-\frac{1}{2|S_2|} & (v \in S_2), \\
0 & (\text{otherwise}).
\end{cases}
$$

Theorem 3.2. *For any graph G*

$$b(G) = \frac{1}{2}\min\{\xi(S_1) + \xi(S_2) : (S_1, S_2) \text{ is a quasi-bipartition of } V\}. \tag{9}$$

Moreover, when (S_1, S_2) is optimal in (9), the corresponding vector $x^{(S_1,S_2)}$ is an ℓ_1-Fiedler vector x.

Proof. Let x be an ℓ_1-Fiedler vector. Let $\kappa(x)$ be the number of distinct positive elements in x (i.e., the cardinality of the set of positive components). Choose x (optimal) with $\kappa(x)$ smallest possible. Define

$$S^+ = \{v : x_v > 0\}, \ S^- = \{v : x_v < 0\} \text{ and } S^0 = \{v : x_v = 0\}.$$

Both S^+ and S^- are nonempty. Also define $M_1 = \max_v x_v$, $M_2 = \min\{x_v : x_v > 0\}$ and

$$S_1^+ = \{v : x_v = M_1\}, \ S_*^+ = \{v : M_2 < x_v < M_1\}, \text{ and } S_2^+ = \{v : x_v = M_2\}.$$

We now prove that there is an optimal solution of (2) where all positive x_v's are equal. If $M_1 = M_2$, there is nothing to prove, so assume $M_1 > M_2$. Let ϵ_1 be a "small" number in absolute value and let ϵ_2 satisfy $|S_1^+|\epsilon_1 = |S_2^+|\epsilon_2$, i.e., $\epsilon_2 = \epsilon_1|S_1^+|/|S_2^+|$. Define x^ϵ by

$$x_v^\epsilon = \begin{cases} M_1 - \epsilon_1 & (v \in S_1^+), \\ M_2 + \epsilon_2 & (v \in S_2^+), \\ x_v & \text{(otherwise)}. \end{cases}$$

Observe that the relationship between ϵ_1 and ϵ_2 assures that

$$\sum_{v:x_v^\epsilon \geq 0} x_v^\epsilon = 1/2 \text{ and } \sum_{v:x_v^\epsilon \leq 0} x_v^\epsilon = -1/2.$$

Thus, $x^\epsilon \in F_1$ provided that ϵ_1 is small enough in absolute value.

Let $\Delta(\epsilon) = f_1(x^\epsilon) - f_1(x)$. Then $\Delta(\epsilon) = \sum_{uv \in E} \Delta_{uv}$ where $\Delta_{uv} = |x_u^\epsilon - x_v^\epsilon| - |x_u - x_v|$ is given as follows for each edge uv
  (a) if $u \in S_1^+$, $v \notin (S_1^+ \cup S_2^+)$, then $\Delta_{uv} = -\epsilon_1$,
  (b) if $u \in S_1^+$, $v \in S_2^+$, then $\Delta_{uv} = -\epsilon_1 - \epsilon_2$,
  (c) if $u \in S_2^+$, $v \in S_*^+$, then $\Delta_{uv} = -\epsilon_2$,
  (d) if $u \in S_2^+$, $v \in S^- \cup S_0$, then $\Delta_{uv} = \epsilon_2$,
and for all other edges $\Delta_{uv} = 0$. Let N_a, N_b, N_c, N_d be the number of edges in categories (a)–(d), respectively. Then

$$\Delta(\epsilon) = N_a(-\epsilon_1) + N_b(-\epsilon_1 - \epsilon_2) + N_c(-\epsilon_2) + N_d(\epsilon_2).$$

By inserting the expression for ϵ_2 above we obtain

$$\Delta(\epsilon_1) = \eta\epsilon_1,$$

for some number η that depends on N_a, N_b, N_c, N_d, $|S_1^+|$ and $|S_2^+|$. Moreover, there exists an $\epsilon^* > 0$ such that for all ϵ_1 with $|\epsilon_1| < \epsilon^*$, the vector x^ϵ lies in F_1. We must have $\eta = 0$, otherwise we could let ϵ_1 be small enough and with opposite sign as η and then

$$0 > \eta \epsilon_1 = \Delta(\epsilon_1) = f_1(x^\epsilon) - f_1(x).$$

So $f_1(x^\epsilon) < f_1(x)$ which contradicts the optimality of x. Therefore, $\eta = 0$ and

$$f_1(x^\epsilon) = f_1(x).$$

Now, let $\epsilon_1 > 0$ and increase ϵ_1 until Δ_{uv} becomes 0 for some edge for which it was previously positive. This happens if either the smallest value in S^+ has been decreased to 0, or when largest value has been decreased to the second largest, or the smallest value has been increased to the second smallest (or both of these occur simultaneously). This x^{ϵ_1} is also an ℓ_1-Fiedler vector. As $\kappa(x^{\epsilon_1}) < \kappa(x)$, this contradicts our choice of x. Thus, by contradiction, it follows that $M_1 = M_2$, so all positive components in x are equal.

Finally, among all ℓ_1-Fiedler vectors whose positive components coincide with that of x, we proceed to treat the negative components in exactly the same manner as the first part of the proof. As a result, we find an ℓ_1-Fiedler vector where all the negative components have the same value, and all the positive components have the same value. Let x denote this vector and define $S_1 = \{v : x_v > 0\}$, $S_2 = \{v : x_v < 0\}$ and $S_0 = \{v : x_v = 0\}$. Let $[S_i : S_j]$ be the set of edges uv such that $u \in S_i, v \in S_j$, where $i, j \in \{0, 1, 2\}$. Then $x_v = 1/(2|S_1|)$ for all $v \in S_1$ and $x_v = -1/(2|S_2|)$ for all $v \in S_2$. Moreover,

$$
\begin{aligned}
b(G) \ &= f_1(x) \\
&= \sum_{uv \in E} |x_u - x_v| \\
&= \sum_{uv \in \delta(S_1) \cup \delta(S_2)} |x_u - x_v| \\
&= \sum_{uv:u \in S_1, v \in S_2} |x_u - x_v| + \sum_{uv:u \in S_1, v \in S_0} |x_u - x_v| + \sum_{uv:u \in S_2, v \in S_0} |x_u - x_v| \\
&= |[S_1 : S_2]|((1/(2|S_1|)) - (-1/(2|S_2|))) + |[S_1 : S_0]|(1/(2|S_1|)) \\
&\quad + |[S_2 : S_0]|(-1/(2|S_2|)) \\
&= (|[S_1 : S_2]| + |[S_1 : S_0]|)((1/(2|S_1|)) - (|[S_1 : S_2]| + |[S_2 : S_0]|)(-(1/(2|S_2|))) \\
&= (1/2)\big(|\delta(S_1)|/|S_1| + |\delta(S_2)|/|S_2|\big) \\
\\
&= (1/2)\big(\xi(S_1) + \xi(S_2)\big).
\end{aligned}
$$

From this calculation we also see that for any $S_1', S_2' \subset V$ with $S_1' \cap S_2' = \emptyset$, $S_1', S_2' \neq \emptyset$, there exists an $x' \in F_1$ with $f_1(x') = (1/2)\big(\xi(S_1') + \xi(S_2')\big)$, and therefore $f_1(x') \geq b(G)$. This proves the theorem. $\square$

Let Γ be the set of all quasi-partitions of V. Thus

$$b(G) = (1/2) \min_{(S_1, S_2) \in \Gamma} \big(\xi(S_1) + \xi(S_2)\big).$$

The next corollary says that if x is an ℓ_1-Fiedler vector then x has no component equal to zero.

Corollary 3.3. *Let $x = (x_1, x_2, \ldots, x_n)$ be an ℓ_1-Fiedler vector. Then $x_i \neq 0$ for all $i \leq n$.*

Proof. Choose an ℓ_1-Fiedler vector x. Partition the set of vertices into $S^+ = \{v : x_v > 0\}$, $S^- = \{v : x_v < 0\}$ and $S^0 = \{v : x_v = 0\}$. We recall that S^+ and S^- are nonempty and we want to prove that $S^0 = \emptyset$. Consider the following edge sets

$$E_1 = \{uv : u \in S^+, v \in S^0\},$$
$$E_2 = \{uv : u \in S^+, v \in S^-\},$$
$$E_3 = \{uv : u \in S^-, v \in S^0\}.$$

Then

$$\begin{aligned} b(G) \ &= \tfrac{1}{2}\big(\xi(S^+) + \xi(S^-)\big) \\ &= \tfrac{1}{2}\big(\tfrac{|\delta(S^+)|}{|S^+|} + \tfrac{|\delta(S^-)|}{|S^-|}\big) \\ &= \tfrac{1}{2}\big(\tfrac{|E_1|+|E_2|}{|S^+|} + \tfrac{|E_2|+|E_3|}{|S^-|}\big). \end{aligned}$$

Assume that $S^0 \neq \emptyset$.

- Let $S' = S^+ \cup S^0$ and consider the quasi-bipartition (S', S^-). Then

$$\xi(S') + \xi(S^-) = \frac{|E_2| + |E_3|}{|S^+| + |S^0|} + \frac{|E_2| + |E_3|}{|S^-|}.$$

  If $|E_3| \leq |E_1|$, then $\frac{1}{2}(\xi(S') + \xi(S^-)) < \frac{1}{2}(\xi(S^+) + \xi(S^-))$, which is not possible by the definition of $b(G)$. Therefore $|E_3| > |E_1|$.
- Let $S'' = S^- \cup S^0$ and consider the quasi-bipartition (S^+, S''). Then

$$\xi(S^+) + \xi(S'') = \frac{|E_1| + |E_2|}{|S^+|} + \frac{|E_1| + |E_2|}{|S^-| + |S^0|}.$$

  If $|E_1| \leq |E_3|$, then $\frac{1}{2}(\xi(S^+) + \xi(S'')) < \frac{1}{2}(\xi(S^+) + \xi(S^-))$, which is not possible by definition of $b(G)$. Therefore $|E_3| < |E_1|$.

Thus, $|E_3| < |E_1| < |E_3|$; a contradiction. We conclude that $S^0 = \emptyset$, as desired. $\square$

The next theorem sums up the results above. It connects $b(G)$ to the minimum edge density and also shows that we may restrict to connected subgraphs when computing $b(G)$.

Theorem 3.4. *For any graph G of order n*

$$b(G) = \frac{n}{2} \min_S \rho(S), \tag{10}$$

where $\rho(S) = \frac{|\delta(S)|}{|S|(n-|S|)}$ is the edge density of the cut $\delta(S)$ and the minimum is taken for nonempty subsets S of V such that $S \neq V$ and both S and its complement induce connected subgraphs of G.

Proof. This follows by combining Theorem 3.2 and Corollary 3.3: we have that the quasi-bipartition (S_1, S_2) must be a partition, i.e., $S_1 \cup S_2 = V$ as from Corollary 3.3 any ℓ_1-Fiedler vector has no components equal to zero. Therefore $\delta(S_1) = \delta(S_2)$. The argument in the proof of Theorem 3.1 assures connectedness of the two subgraphs. The definition of edge density of a cut and the relation in (4) give the desired formula for $b(G)$. $\square$

Thus we arrive at the important insight:

- *up to a multiplicative constant, namely $n/2$, the optimal value $b(G)$ in the ℓ_1-graph smoothing problem coincides with the smallest edge density of a cut in G.*

This gives a very intuitive interpretation of the partitioning of a graph G according to the positive and negative values in the ℓ_1-Fiedler vector: it corresponds to a cut of smallest edge density, also called a *sparsest cut*. In fact, for such a sparsest cut $\delta(S)$ a corresponding ℓ_1-Fiedler vector is $x = x^{(S,V\setminus S)} = (x_v : v \in V)$ given by

$$x_v = \begin{cases} \frac{1}{2|S|} & (v \in S), \\ -\frac{1}{2|\bar{S}|} & (v \in V \setminus S). \end{cases} \tag{11}$$

Thus, this correspondence is underlying when we later refer to a solution of the ℓ_1-graph smoothing problem or the sparsest cut problem, i.e., one solution can be converted into the other.

Example 2. Consider the path $P_5 = v_1, v_2, v_3, v_4, v_5$. By Theorem 3.4, including the connectivity result, it is easy to see that a sparsest cut is $\delta(S^*)$ where $S^* = \{v_1, v_2\}$. So $\min_S \rho(S) = 1/6$ and $b(G) = 5/12$.

Example 3. The cube graph is the graph formed by the 8 vertices and 12 edges of a three-dimensional cube. For the cube graph $\min_S \rho(S) = 1/4$ so $b(G) = 1$.

Due to this close connection to the sparsest cut, we can now conclude that computing an ℓ_1-Fiedler vector is a computationally hard problem.

Corollary 3.5. *The computation of $b(G)$ and a corresponding ℓ_1-Fiedler vector is* NP-*hard.*

Proof. It was shown in [7] that computing $\min\{\rho(S) : \emptyset \subset S \subset V\}$ is *NP*-hard. Therefore, due to Theorem 3.4, the desired conclusion follows. □

As mentioned, in [7] it is shown that the sparsest cut is *NP*-hard. The paper further contains a number of results showing that the problem is polynomially solvable for certain classes of graphs. For instance, this applies to graphs with bounded treewidth. Furthermore, for cactus graphs, i.e., connected graphs where each edge is in at most one cycle, there is an algorithm for finding a sparsest cut which is linear in n, the number of vertices. This class includes trees, which we return to below. In [6] an efficient approximation algorithm for the sparsest cut was established, and it gives an $O(\sqrt{\log n})$ approximation (to the minimum value). In [30] there is whole chapter devoted to the sparsest cut problem. Here the connection to a certain linear programming problem is shown. This is a multicommodity network flow problem where the goal is to maximize throughput for a given set of demands given by origin/destination (OD-) pairs and the corresponding flow demand. The special case where all demands are 1 and every pair is an OD-pair leads to an upper bound of the throughput which is the edge density. Therefore the minimum edge density, and a sparsest cut, corresponds to a bottleneck of the multicommodity flow problem. In this approach linear programming duality is combined with some basic results on embeddability of ℓ_1-metric spaces. This leads to a very important approximation algorithm with approximation error $O(\log n)$; for the details we refer to [30] or [28].

Sparsest cuts are used in applications. For instance, in [31] sparsest cuts (which is called ratio cuts there) are used in image segmentation. They also show that the sparsest cut problem is polynomially solvable in planar graphs, which is of interest in image analysis. In [24] one considered a segmentation (or decomposition) problem in image analysis, using a model based on partial differential equations and total variation norm (L_1-norm).

4. Comparison with other parameters

This section establishes bounds on $b(G)$ in terms of other parameters.

By combining Theorem 3.4 and Theorem 1.1 we now obtain the following bounds on $b(G)$.

Corollary 4.1. *Let G be a graph of order n. Then*

$$(1/2)a(G) \leq b(G) \leq (1/2)\lambda_1,$$

where $a(G)$ is the algebraic connectivity of G and λ_1 is the largest eigenvalue of $L(G)$.

Let $d_{\min}(G)$ denote the smallest vertex degree in G.

Corollary 4.2. *For any graph G of order n the following bounds on $b(G)$ hold*

$$\min_S \xi(S) \leq b(G) \leq \frac{n}{2(n-1)} d_{\min}(G).$$

Proof. Let S_1 and S_2 be such that the minimum in (9) is attained (so $S_2 = V \setminus S_1$). Then

$$\min_S \xi(S) \leq \min\{\xi(S_1), \xi(S_2)\} \leq (1/2)(\xi(S_1) + \xi(S_2)) = b(G).$$

Next, let v be a vertex with smallest degree in G, so $d_v = d_{\min}(G)$, and let $S = \{v\}$. Then $\rho(S) = d_v/(n-1)$ so, by Theorem 3.2, an upper bound on $b(G)$ is $\frac{n}{2(n-1)} d_{\min}(G)$. □

Let $mc(G)$ denote the cardinality of a minimum cut in G, i.e.,

$$mc(G) = \min\{|\delta(S)| : \emptyset \subset S \subset V\}.$$

The next result is proved similar to the upper bound in the previous corollary, by letting S be such that $\delta(S)$ is a minimum cut.

Corollary 4.3. *For any graph G*

$$b(G) \leq \frac{n}{2s(n-s)} mc(G),$$

where $s = |S|$ and $\delta(S)$ is a minimum cut in G.

This bound is of interest because a minimum cut may be found efficiently (polynomial time) by simple greedy algorithms. Moreover, such a minimum cut may be the starting point of algorithms for computing approximations to $b(G)$.

Let $m = |E|$ be the number of edges in G.

Theorem 4.4. *For every graph G*

$$b(G) \leq \sqrt{m \cdot a(G)}. \tag{12}$$

Proof. Let x' be a Fiedler vector of unit length, so $\sum_{uv \in E}(x'_u - x'_v)^2 = a(G)$ and $\sum_v x'_v = 0$, $\|x'\|_2 = 1$. Let $x^* = tx'$ where $t = 1/\|x'\|_1$. Then $\|x^*\|_1 = 1$, $\sum_j x^*_j = 0$ and $t = 1/\|x'\|_1 \leq 1/\|x'\|_2 = 1$ as, in general, $\|z\|_2 \leq \|z\|_1$ for $z \in \mathbb{R}^n$. Therefore

$$
\begin{aligned}
b(G) \quad &= \min\{\textstyle\sum_{ij\in E} |x_i - x_j| : \sum_j x_j = 0, \ \|x\|_1 = 1\} \\
&\leq \textstyle\sum_{ij\in E} |x_i^* - x_j^*| \\
&= t \textstyle\sum_{ij\in E} |x_i' - x_j'| \cdot 1 \\
&\leq (\textstyle\sum_{ij\in E}(x_i' - x_j')^2)^{1/2} \cdot (\sum_{ij\in E} 1^2)^{1/2} \\
&= \sqrt{m \cdot a(G)},
\end{aligned}
$$

where the last inequality is obtained from the Cauchy-Schwarz inequality. □

5. Examples and special graphs

This section contains examples and results for special graphs.

The next example considers the complete graph, which is an extreme case in terms of the parameter $b(G)$.

Example 4. Consider K_n the complete graph of order n. Let $S \subset V$ where $S \neq \emptyset, V$. Define $s = |S|$. Then

$$
\rho(S) = \frac{s(n - s)}{s(n - s)} = 1,
$$

which is independent of S. So, by Theorem 3.4, $b(K_n) = \frac{n}{2}$. Note that K_n is an $(n - 1)$-regular graph. It is known that $a(K_n) = n$. Thus $b(K_n) < a(K_n)$.

Example 5. Consider the cycle C_n, with $n \geq 4$ even. For all S, such that $|S| = p$, then $\rho(S) = \frac{2}{p(n-p)}$. In this case $\min_S \rho(S) = \frac{8}{n^2}$, so $b(G) = 4/n$. If $n \geq 3$ and n is odd, then $\min_S \rho(S) = \frac{2}{\lfloor n/2 \rfloor \cdot \lceil n/2 \rceil}$ and $b(G) = \frac{n}{\lfloor n/2 \rfloor \cdot \lceil n/2 \rceil}$.

A wheel graph W_n, with $n \geq 4$, is a graph formed by connecting a single vertex (central vertex) to all vertices of a cycle.

Example 6. Let $n \geq 4$. Then $b(W_n) = \frac{n}{n-2}$. In fact,

1. Consider all subsets S, such that $|S| = 1$, say $S = \{v\}$. Then, two situations can occur:
   - v is the central vertex. Then, $\rho(S) = \frac{n-1}{n-1} = 1$.
   - v is not the central vertex. Then, $\rho(S) = \frac{3}{n-1}$.
2. Consider all remaining subsets S, such that $|S| = i \geq 2$. Again, two situations can occur:
   - S does not contain the central vertex. Then, $\rho(S) = \frac{i+2}{i(n-i)}$.
   - S contains the central vertex. Then $\rho(S) = \frac{n-i+2}{i(n-i)}$.

With some calculus we see that $\min_S \rho(S)$ is equal to $2/(n-2)$, so $b(G) = n/(n-2)$.

For trees we can find a sparsest cut analytically, as described next. Let $T = (V, E)$ be a tree, and let $e = uv \in E$. Then $T \setminus \{e\}$ consists of two disjoint trees, T_u^e and T_v^e, where $T_u^e = (V_u^e, E_u^e)$ contains u, $T_v^e = (V_v^e, E_v^e)$ contains v and $V_u^e \cup V_v^e = V$. We call $e = uv$ a *center edge* if $||V_u^e| - |V_v^e||$ is smallest possible and we let $\Delta(T)$ denote this minimum value. For instance, let T be the path P_n. If $n = |V|$ is even, then $\Delta(P_n)$ is 0, and there exist only one center edge. If n is odd, then $\Delta(P_n)$ is 1 and there are two center edges. If T is the star S_n, then $\Delta(S_n) = n - 2$ and all the edges of S_n are center edges.

Theorem 5.1. *Let $T = (V, E)$ be a tree. Then, there exists a center edge $e = uv \in E$ such that the cut $\delta(V_u^e)$ is a sparsest cut, and*

$$b(T) = (1/2)\big(1/|V_u^e| + 1/|V_v^e|\big).$$

Proof. We apply Theorem 3.4, so a sparsest cut must be of the form $C = \delta(V_u^e)$ for some edge $e = uv$. The corresponding edge density of the cut C is

$$\rho(V_u^e) = \frac{1}{|V_u^e||V_v^e|},$$

and it is easy to compute that this is minimized precisely when $||V_u^e| - |V_v^e||$ is smallest possible, i.e., when e is a center edge and therefore

$$b(T) = (n/2)\rho(V_u^e) = (1/2)\big(1/|V_u^e| + 1/|V_v^e|\big). \quad \square \tag{13}$$

A *substar* S in a tree T is a vertex-induced subgraph which is a star, i.e., it consists of edges sharing a common vertex.

Corollary 5.2. *Let $T = (V, E)$ be a tree. Then the set of center edges is a substar.*

Proof. We prove this by contradiction. So, assume there are two center edges $e = uv$ and $e' = u'v'$ that are disjoint (no common vertex). Since T is connected there is a path P between a vertex in e and a vertex in e', say that v and v' are closest with this property.

Claim: $|V_u^e| \leq |V_v^e|$, and $|V_{u'}^{e'}| \leq |V_{v'}^{e'}|$.

Proof of Claim: Otherwise $|V_u^e| > |V_v^e|$, and $V_{u'}^{e'} \subset V_v^e$ (strict subset) so $|V_{v'}^{e'}| > |V_u^e| > |V_v^e| > |V_{u'}^{e'}|$. This contradicts that e' is a center edge (because the two numbers to the left and right in these inequalities are further apart that the two in the middle). This proves the claim.

Next, from the Claim, we conclude that $|V_u^e| = |V_{u'}^{e'}|$ because both e and e' are center edges (otherwise these two numbers would have different distance to $n/2$; see the discussion on center edge above). Let $e^* = vw$ be the edge in the path P that is incident to v. Then e^* is different from e and e'. Moreover, $|V_v^{e^*}| \geq n/2$, otherwise $|V_v^{e^*}|$

would be closer to $n/2$ than $|V_u^e|$; contradicting that e is a center edge. This implies that $|V_w^{e^*}| \leq n/2$ and then

$$|V_{u'}^{e'}| < |V_w^{e^*}| \leq n/2,$$

contradicting that e' is a center edge. Therefore, such an edge e^* does not exist, meaning that e and e' must be incident. $\square$

Corollary 5.3. *Let P_n be the path with n vertices.*

If n is even, then $b(P_n) = 2/n$ and an ℓ_1-Fiedler vector has the first $n/2$ components equal to $1/n$ and the last $n/2$ components are equal to $-1/n$.

If n is odd, then $b(P_n) = 2n/(n^2 - 1)$. An ℓ_1-Fiedler vector has the first $(n-1)/2$ components equal to $1/(n-1)$ and the remaining components are equal to $-1/(n+1)$.

Proof. This follows from Theorem 5.1. Let n be even. Taking $|V_u^e| = p$, and $|V_v^e| = n - p$, we have $\delta(P_n) = \frac{1}{p(n-p)}$, where the minimum of this function is attained for $p = n/2$. Then $b(P_n) = 2/n$. From the proof of Theorem 3.2 the ℓ_1-Fiedler vector has $p = n/2$ entries equal to $\frac{1}{2p} = \frac{1}{n}$ and $n/2$ entries equal to $-\frac{1}{2(n-p)} = -\frac{1}{n}$. The case when n is odd is analogous. $\square$

Corollary 5.4. *Let S_n be the star with n vertices. Then*

$$b(S_n) = \frac{1}{2} + \frac{1}{2(n-1)}.$$

Moreover, the entries of an ℓ_1-Fiedler vector are $\frac{1}{2}$ for the center vertex and $-\frac{1}{2(n-1)}$ for the remaining vertices.

Proof. This follows from Theorem 5.1 and Theorem 3.2. $\square$

Let $\mathcal{C}$ be the set of center edges of a given tree. We next look at some examples concerning specific classes of trees.

Example 7. Consider a *broom-tree* $B(\ell, n - \ell)$ consisting of a path $v_1, v_2, \ldots, v_\ell$ and additional vertices $v_{\ell+1}, v_{\ell+2}, \ldots, v_n$ each attached to v_ℓ. Here e_{ij} denotes the edge $v_i v_j$ or simply ij. There are two cases to discuss.

1. If $\ell \leq n - \ell$, i.e., $\ell \leq n/2$, then there is a unique center edge, and it is $e = e_{\ell-1,\ell}$. So $\mathcal{C} = \{e\}$ and

$$b(B(\ell, n - \ell)) = \frac{1}{2}\Big(\frac{1}{\ell - 1} + \frac{1}{n - \ell + 1}\Big) = \frac{n}{2(\ell - 1)(n - \ell + 1)}. \tag{14}$$

In fact, let us consider an edge $e_{k,k+1}$ with $k = 1, 2, \ldots, \ell - 1$, and the function $f(k) = \frac{1}{2}(\frac{1}{k} + \frac{1}{n-k})$. This function is decreasing and its minimum is attained for

$k = n - 1$. Additionally, for any edge $e_{\ell,\ell+p}$ for $p = 1, \ldots, n - p$, the expression in (13) is $(1/2)\left(1 + \frac{1}{n-1}\right)$ that is clearly greater or equal than the expression in (14). Thus $b(B(\ell, n - \ell))$ is as in (14).

2. If $\ell > n - \ell$, i.e., $\ell > n/2$, then two subcases can occur.

   (a) n is even, say $n = 2k$ for some integer k.

   Then, $e = e_{k,k+1}$ is the unique center edge, so $\mathcal{C} = \{e\}$. In this case we have

   $$|V_{v_k}^e| = |V_{v_{k+1}}^e| = \frac{n}{2},$$

   and

   $$b(B(\ell, n - \ell)) = \frac{1}{2}\left(\frac{1}{n/2} + \frac{1}{n/2}\right) = \frac{2}{n} = \frac{1}{k}.$$

   (b) n is odd, say $n = 2k + 1$ for some integer k.

      i. If $\ell - k = 1$, then there exists a unique center edge $e = e_{\frac{n-1}{2}, \frac{n+1}{2}}$ and $\mathcal{C} = \{e\}$. Then

      $$b(B(\ell, n - \ell)) = \frac{1}{2}\left(\frac{1}{k} + \frac{1}{n-k}\right) = \frac{1}{n-1} + \frac{1}{n+1} = 2n/(n^2 - 1). \quad (15)$$

      ii. Suppose now that $\ell - k > 1$. Then we have two center edges $e = e_{k,k+1}$ and $e^\star = e_{k+1,k+2}$, that is $e = e_{\frac{n-1}{2}, \frac{n+1}{2}}$ and $e^\star = e_{\frac{n+1}{2}, \frac{n+3}{2}}$. In this case $\mathcal{C} = \{e, e^\star\}$ and $b(B(\ell, n - \ell))$ is the same value as (15).

A pendant edge in a graph is an edge where one of its vertices has degree 1.

Example 8. Let $n_1, n_2, \ldots, n_k$ be positive integers. The *starlike tree* $S(n_1, n_2, \ldots, n_k)$ is the tree that results from the stars $S_{n_1}, S_{n_2}, \ldots, S_{n_k}$ by connecting their centers to an extra vertex v, see [19]. Let $n = \sum_{i=1}^k n_i + 1$ and define $n_M = \max_{i=1,2,\ldots,k} n_i$. Then,

$$b(S(n_1, n_2, \ldots, n_k)) = \frac{n}{2n_M(n - n_M)}.$$

If $|S_p| > |S_i|$ ($i \neq p$), then there is a unique center edge, which is vv_ℓ. Otherwise the maximum of $|S_i|$ is attained for more than one i, and the center edges constitute a substar connecting v to the center of those stars S_i.

Finally, in this section, we give a computational example for a graph G. It illustrates the partitions obtained based on the Fiedler vector and the ℓ_1-Fiedler vector, respectively.

Example 9. In this example we generated some "random" points in the plane and constructed edges between points that were closer than some given distance. This gave a graph G. The graph contained 15 vertices and 69 edges. We wanted to compare the partitions obtained from the ℓ_2 (Fiedler) versus ℓ_1 (sparsest cut) graph smoothing approach.
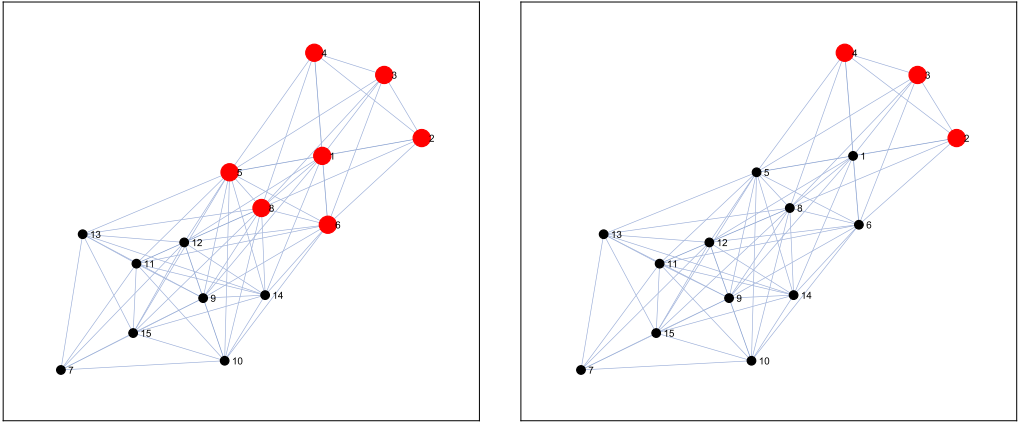
Fig. 2. Graph smoothing partitions: ℓ_2 (left) and ℓ_1 (right).

In the ℓ_2 approach we obtained a cut with 22 edges and the partition contained 7 and 8 vertices, respectively, see Fig. 2. The two vertex sets in the partition are indicated with different bullet sizes. In the ℓ_1 approach we obtained a cut with 10 edges and the partition contained 3 and 12 vertices, respectively. The corresponding edge densities were 0.39 (for ℓ_2) and 0.28 (for ℓ_1). So the minimum edge density is 0.28. Note that the sparsest cut is quite unbalanced, but has very few edges compared to the cut in the Fiedler partition. For certain graphs this may happen, but graphs are so different that we will not make any general claims on properties of these solutions.

Concerning the previous example, note that each of the two solutions are optimal in the respective optimization problems, so comparing partitions is a bit artificial. However, both methods may be used for partitioning, and which one is the better is hard to say in general. It depends on the underlying application and the type of graph considered. Computationally, the Fiedler vector can be computed fast, while for larger graphs we need heuristics for finding approximate sparsest cuts. On the other hand, a sparse or sparsest cut is a reasonable notion and can be analyzed for specific graph classes, while the Fiedler vector relies on an eigenvector which is not easy to give a direct combinatorial interpretation. Finally, for clustering it might be possible to combine these two methods, possibly also with some of the many other approaches known for graph clustering.

6. Graph smoothing in ℓ_∞-norm and concluding remarks

We conclude the paper with some remarks. First, it also makes sense to consider the graph smoothing problem in other norms, such as the ℓ_∞-norm (given by $\|x\|_\infty = \max_i |x_i|$). This leads to the ℓ_∞-*graph smoothing* problem

$$\gamma(G) = \min\{\max_{uv\in E} |x_u - x_v| : \sum_{v\in V} x_v = 0, \ \|x\|_\infty = 1\}. \tag{16}$$

In contrast to the ℓ_1-graph smoothing problem we can solve (16) efficiently, for a general graph G.

Theorem 6.1. *The ℓ_∞-graph smoothing problem* (16) *can be solved in polynomial time, using linear programming.*

Proof. We rewrite problem (16) using a construction rather similar to what we did for the ℓ_1-graph smoothing problem in (7). For each $k \leq n$ consider the linear programming problem LP(k):

$$
\begin{array}{rrcll}
\text{minimize} & y & & & \\
& x_i - x_j & \leq & y & (ij \in E), \\
& -(x_i - x_j) & \leq & y & (ij \in E), \\
& \sum_{j=1}^{n} x_j & = & 0, & \\
& -1 \leq x_j \leq 1 & & (j \leq n), & \\
& x_k = 1. & & &
\end{array}
\tag{17}
$$

Here the variables are $x = (x_1, x_2, \ldots, x_n)$ and y. Note that in every optimal solution of (17) $y = \max_{uv \in E} |x_u - x_v|$, and $\|x\|_\infty = 1$. It follows that LP(k) solves the ℓ_∞-graph smoothing problem under the additional restriction that $x_k = 1$. (Due to symmetry, $-x$ satisfies the first four constraints when x does, so restricting one component to -1 is not needed.) So, by solving LP(k) for every $k \leq n$, and taking the minimum y found, we solve the ℓ_∞-graph smoothing problem. Since LP problems can be solved in polynomial time, the resulting algorithm is a polynomial-time algorithm. $\square$

Proposition 6.2. *Consider the path P_n with n vertices. Then the minimum value of the ℓ_∞-graph smoothing problem* (16) *is $\gamma = \gamma(P_n) = 2/(n-1)$ and an optimal solution is*

$$
x = (1, 1 - \gamma, 1 - 2\gamma, \ldots, 1 - (n-2)\gamma, -1).
$$

Proof. For the specified x we have $|x_i - x_{i+1}| = \gamma$ for $i \leq n-1$, so $\max_i |x_i - x_{i+1}| = \gamma$. Moreover, $\sum_i x_i = 0$ and $\|x\|_\infty = 1$. Assume there is an $z = (z_1, z_2, \ldots, z_n)$ satisfying $\sum_i z_i = 0$, $\|z\|_\infty = 1$ and $\max_i |z_i - z_{i+1}| < \gamma$. By symmetry we may assume that some component of z is 1, say $z_k = 1$. Then, if $k < n$, $|z_k - z_{k+1}| < \gamma$, so $z_{k+1} > 1 - \gamma$. Similarly, $z_{k+2} > 1 - 2\gamma$, and in general $z_s > 1 - |s - k|\gamma$. But then it is easy to see that

$$
\sum_i z_i > \sum_i x_i = 0,
$$

which contradicts that $\sum_i z_i = 0$. Thus, the optimal value of (16) is not less than γ, and therefore equal to γ, as desired. $\square$

Example 10. Consider the path $P_6 = v_1, v_2, v_3, v_4, v_5, v_6$, with corresponding variables x_i ($i \leq 5$). Then $\gamma = 2/5$ and an optimal solution of the ℓ_∞-graph smoothing problem is

$$x = (1, 3/5, 1/5, -1/5, -3/5, -1).$$

Thus consecutive components of x differ in absolute value by $y = 2/5$.

One can also solve (16) explicitly for stars, and some other graphs. It is ongoing work to investigate the ℓ_∞-graph smoothing problem further and to relate to the ℓ_1- and ℓ_2-graph smoothing problems. Moreover, it is interesting to see if the corresponding cuts obtained from the signs in an optimal x^* are useful in partitioning problems.

Finally, we remark that the main contribution of the present paper was to investigate a variant of the optimization (variational) characterization of algebraic connectivity, by changing into the ℓ_1-norm. We showed strong optimality properties that are similar to the Fiedler theory for algebraic connectivity. Also, we showed that optimal solutions correspond to sparsest cuts which gives a new way to view these combinatorial objects. We believe that further work on similar optimization problems, in different norms, would be interesting. This includes to combine the different approaches into useful algorithms for important applications in clustering problems in graphs.

Recently, in the revision process, we learned about a PhD thesis (in Portuguese) by E.M. Borba [8]. Here the author treats the so-called (p, q)-algebraic connectivity which is a parameter $a_p^q(G)$ associated with the q-Laplacian L_q defined in [3]. For $p = q = 1$ this leads to a minimization problem which is related, but different, from our ℓ_1-graph smoothing problem; the difference is that we also use the constraint that x is orthogonal to the all ones vector. It was shown in [8] that $a_1^1(G)$ is equal to Cheeger's constant.

Declaration of competing interest

There is no competing interest.

Data availability

No data was used for the research described in the article.

Acknowledgements

References

[1] N. Alon, Eigenvalues and expanders, Combinatorica 6 (1986) 83–96.
[2] N. Alon, A. Milman, Isoperimetric inequalities for graphs, and superconcentrators, J. Comb. Theory, Ser. B 38 (1985) 73–88.
[3] S. Amghibech, Eigenvalues of the discrete p-Laplacian for graphs, Ars Comb. 67 (2003) 283–302.
[4] E. Andrade, L. Ciardo, G. Dahl, Combinatorial Perron parameters for trees, Linear Algebra Appl. 566 (2019) 138–166.
[5] E. Andrade, G. Dahl, Combinatorial Perron values of trees and bottleneck matrices, Linear Multilinear Algebra 65 (12) (2017) 2387–2405.
[6] S. Arora, E. Hazan, S. Kale, $O(\sqrt{\log n})$ approximation to sparsest cut in $\tilde{O}(n^2)$ time, SIAM J. Comput. 39 (5) (2010) 1748–1771.
[7] P. Bonsma, H. Broersma, V. Patel, A. Pyatkin, The complexity of finding uniform sparsest cuts in various graph classes, J. Discret. Algorithms 14 (2012) 136–149.
[8] E. Max Borba, Generalizações analíticas do espetro de um grafo, Ph. D. thesis, Universidade Federal do Rio Grande do Sul, Instituto de Matemática, Brazil, 2018.
[9] L. Ciardo, S. Živný, Approximate graph colouring and crystals, in: Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), 2023, pp. 2256–2267.
[10] J. Cheeger, A lower bound for the smallest eigenvalue of the Laplacian, in: R.C. Gunnig (Ed.), Problems in Analysis, Princeton Univ. Press, Princeton, NJ, 1970, pp. 195–199.
[11] R.W. Cottle, G.B. Dantzig, Complementary pivot theory of mathematical programming, Linear Algebra Appl. 1 (1968) 103–125.
[12] S. Hoory, N. Linial, A. Wigderson, Expander graphs and their applications, Bull. Amer. Soc. 43 (4) (2006) 439–561.
[13] S. Fallat, S. Kirkland, S. Pati, On graphs with algebraic connectivity equal to minimum edge density, Linear Algebra Appl. 373 (2003) 31–50.
[14] M. Fiedler, Algebraic connectivity of graphs, Czechoslov. Math. J. 23 (2) (1973) 298–305.
[15] M. Fiedler, A property of eigenvectors of nonnegative symmetric matrices and its application to graph theory, Czechoslov. Math. J. 25 (1975) 619–633.
[16] R.A. Horn, C.R. Johnson, Matrix Analysis, second edition, Cambridge University Press, New York, 2013.
[17] M. Juvan, B. Mohar, Optimal linear labelling and eigenvalues of graphs, Discrete Appl. Math. 36 (1992) 153–168.
[18] S. Kim, S. Kirkland, Fiedler vectors with unbalanced sign patterns, Czechoslov. Math. J. 71 (146) (2021) 1071–1098.
[19] A. Knopfmacher, R.F. Tichy, S. Wagner, V. Ziegler, Graphs, partitions and Fibonacci numbers, Discrete Appl. Math. 155 (10) (2007) 1175–1187.
[20] U. von Luxburg, A tutorial on spectral clustering, Stat. Comput. 17 (4) (2007) 395–416.
[21] B. Mohar, Laplace eigenvalues of graphs – a survey, Discrete Math. 109 (1992) 171–183.
[22] B. Mohar, Isoperimetric numbers of graphs, J. Comb. Theory, Ser. B 47 (1989) 274–291.
[23] J.J. Molitierno, Applications of Combinatorial Matrix Theory to Laplacian Matrices of Graphs, CRC Press, Boca Raton, 2012.
[24] S. Osher, A. Solé, L. Vese, Image decomposition and restoration using total variation minimization and the H^{-1} norm, SIAM Multiscale Model. Simul. 1 (3) (2003) 349–370.
[25] S. Schaeffer, Graph clustering, Comput. Sci. Rev. 1 (1) (2007) 27–64.
[26] D.A. Spielman, Algorithms, graph theory, and linear equations in Laplacian matrices, in: Proceedings of the International Congress of Mathematicians, Hyderabad, India, 2010.
[27] D.A. Spielman, Spectral and Algebraic Graph Theory, Lecture Notes, Yale University, 2019.
[28] L. Trevisan, Lecture Notes on Graph Partitioning, Expanders and Spectral Methods, University of California Lecture Notes, Berkeley, 2017.
[29] C. Urschel, L.T. Zikatanov, On the maximal error of spectral approximation of graph bisection, Linear Multilinear Algebra 64 (2016) 1972–1979.
[30] V.V. Vazirani, Approximation Algorithms, Second printing, Springer, Berlin Heidelberg, 2003.
[31] S. Wang, J.M. Siskind, Image segmentation with ratio cut, IEEE Trans. Pattern Anal. Mach. Intell. 25 (6) (2003) 675–690.