Discrete Applied Mathematics 347 (2024) 143–154

Contents lists available at ScienceDirect

Discrete Applied Mathematics

journal homepage: www.elsevier.com/locate/dam

The complexity of spanning tree problems involving graphical indices[☆]

Yanni Dong [a,b,c], Hajo Broersma [c,*], Yuhang Bai [a,b], Shenggui Zhang [a,b]

[a] School of Mathematics and Statistics, Northwestern Polytechnical University, Xi'an, Shaanxi 710129, PR China
[b] Xi'an-Budapest Joint Research Center for Combinatorics, Northwestern Polytechnical University, Xi'an, Shaanxi 710129, PR China
[c] Faculty of Electrical Engineering, Mathematics and Computer Science, University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands

ARTICLE INFO

Article history:
Received 9 November 2022
Received in revised form 1 December 2023
Accepted 10 January 2024
Available online 19 January 2024

Keywords:
Graphical function-index
Spanning tree
NP-complete
APX-complete

ABSTRACT

We consider the computational complexity of spanning tree problems involving the graphical function-index. This index was recently introduced by Li and Peng as a unification of a long list of chemical and topological indices. We present a number of unified approaches to determine the $\mathcal{NP}$-completeness and $\mathcal{APX}$-completeness of maximum and minimum spanning tree problems involving this index. We give many examples of well-studied topological indices for which the associated complexity questions are covered by our results.

1. Introduction

Within the popular area of chemical graph theory, so-called graphical indices (also known as chemical indices or topological indices) play an important role in capturing the structural properties of molecules. Our main focus here is on graphical indices which are based on functions of the degrees of the vertices of the graphs that represent these molecules. Here the degree of a vertex x of a graph G, denoted by $d_G(x)$, is the number of edges of G incident with x. Adopting the terminology of Li and Peng [22], for a symmetric real function $f(x, y)$, we use the unifying term graphical function-index of a graph $G = (V, E)$ for the expression $\sum_{uv\in E} f(d_G(u), d_G(v))$. This definition captures many well-studied graphical indices, some of which we included with their commonly used name in Table 1.

The above term also captures graphical indices defined in [30,33] by a real function $f(x)$ and the expression $\sum_{u\in V} f(d_G(u))$. This follows since $\sum_{uv\in E}(\frac{f(d_G(u))}{d_G(u)} + \frac{f(d_G(v))}{d_G(v)}) = \sum_{u\in V} f(d_G(u))$ (i.e., by choosing $f(x, y) = \frac{f(x)}{x} + \frac{f(y)}{y}$ in the above expression).

A spanning subgraph of a graph G is obtained by only deleting edges of G. A spanning tree of G is a spanning subgraph and a tree (i.e., a connected spanning subgraph without cycles). It is not difficult to figure out (and a folklore result) that a spanning tree of a connected graph G (on n vertices) is a connected spanning subgraph of G with the minimum number of $(n - 1)$ edges. The following figure shows examples of a spanning subgraph and a spanning tree (see Fig. 1).

Spanning tree problems arise in many application areas. As one can see from the above, spanning trees offer the cheapest way (in terms of the number of edges) to connect a number of objects (modelled by vertices of a graph) by

☆ The research was supported by the National Natural Science Foundation of China (Nos. 12071370, 12131013 and U1803263) and China Scholarship Council, China (No. 202006290070).

* Corresponding author.
E-mail address: h.j.broersma@utwente.nl (H. Broersma).

https://doi.org/10.1016/j.dam.2024.01.005
0166-218X/

Table 1
Some known graphical function-indices.

Name	$f(x, y) =$
First Zagreb index	$x + y$
Second Zagreb index	xy
First hyper-Zagreb index	$(x + y)^2$
Second hyper-Zagreb index	$(xy)^2$
Modified first Zagreb index	$x^{-3} + y^{-3}$
Albertson index	$\|x - y\|$
Extended index	$(x/y + y/x)/2$
Sigma index	$(x - y)^2$
Randić index	$1/\sqrt{xy}$
Reciprocal Randić index	$\sqrt{xy}$
Sum-connectivity index	$1/\sqrt{x + y}$
Reciprocal sum-connectivity index	$\sqrt{x + y}$
Harmonic index	$2/(x + y)$
Atom-bond connectivity index	$\sqrt{(x + y - 2)/(xy)}$
Augmented Zagreb index	$x^3y^3/(x + y - 2)^3$
Forgotten index	$x^2 + y^2$
Inverse degree	$x^{-2} + y^{-2}$
Geometric-arithmetic index	$2\sqrt{xy}/(x + y)$
Arithmetic-geometric index	$(x + y)/2\sqrt{xy}$
Inverse sum index	$xy/(x + y)$
First Gourava index	$x + y + xy$
Second Gourava index	$(x + y)xy$
First hyper-Gourava index	$(x + y + xy)^2$
Second hyper-Gourava index	$x^2y^2(x + y)^2$
Sum-connectivity Gourava index	$1/\sqrt{x + y + xy}$
Product-connectivity Gourava index	$\sqrt{(x + y)xy}$

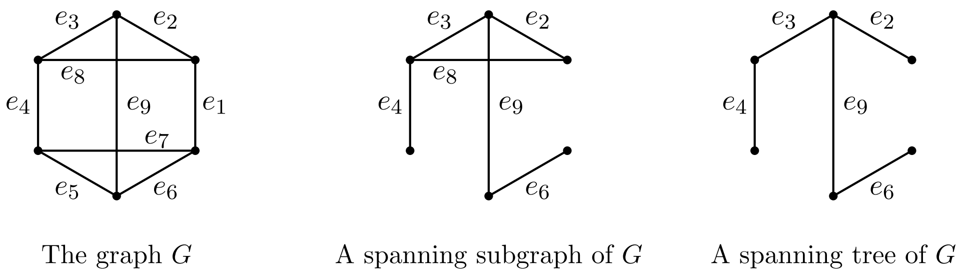

Fig. 1. Some spanning subgraphs of G.

using links (modelled by edges in the graph) between pairs of objects. In more general situations, the links have associated costs, lengths, etc. (modelled by assigning positive weights to the edges), and one is looking for a spanning tree which minimizes the total weight of the edges. On the other hand, if the weights correspond to profits, gains, etc., one is looking for a spanning tree which maximizes the total weight.

Such spanning tree problems turn up naturally in logistics and network applications, but also in less expected domains. We give a few examples, starting with an application in cluster analysis. In [29], minimum spanning trees are used to find gene clusters of various shapes. Similarly, in [6] the authors first construct a minimum spanning tree from local density peaks, and then iteratively replace inconsistent edges until a predetermined number of clusters are obtained.

Our second example arises in remote sensing and space science. In a recent paper [25], the authors used minimum spanning trees to connect peaks in radargrams, in order to analyse and interpret the subsurface structure of Mars.

For solving minimum spanning tree problems, there exist well-known polynomial algorithms, like Kruskal's algorithm and Prim's algorithm, which can be found in any textbook on graph theory, including [3].

In the context of the graphical function-index $\sum_{uv\in E} f(d_G(u), d_G(v))$ we introduced before, it is natural to interpret the weight of each edge $uv \in E$ as $f(d_G(u), d_G(v))$, and consider the complexity of spanning tree problems within this context. In particular, we aim to unify decision problems concerning lower and upper bounds on the value of the graphical function-index of spanning trees, as well as the associated optimization problems, for different choices of the function

$f(x, y)$. For this reason, we assume that all graphs are finite, simple and connected. We refer to [3,13] for any undefined notation and terminology related to graph theory and complexity, respectively.

2. Spanning tree problems and their complexity

In this section, we prove several $\mathcal{NP}$-completeness results. In the first part, we focus on decision problems related to the graphical function-index $\sum_{uv\in E} f(d_G(u), d_G(v))$. In the second part, we continue with some complexity results in case $f(x, y) = \frac{f(x)}{x} + \frac{f(y)}{y}$ for a real function $f(x)$, in which case the above expression for the graphical function-index is equivalent to $\sum_{u\in V} f(d_G(u))$.

2.1. Conditions on extremal trees

We use $GFI_f(G)$ to denote the graphical function-index of a graph $G = (V, E)$, where $GFI_f(G) = \sum_{uv\in E} f(d_G(u), d_G(v))$ for a suitable choice of the symmetric real function $f(x, y)$. Since we deal with complexity questions, throughout the paper we assume that the chosen function $f(x, y)$ is computable in polynomial time, without explicitly mentioning it.

In the following, we are interested in the computational complexity of determining a spanning tree T of G that maximizes (minimizes) $GFI_f(T)$. For this purpose, we define the two associated decision problems as follows.

MAXST-GFI$_f$
INSTANCE: A graph G, and a real number k.
QUESTION: Does G have a spanning tree T with $GFI_f(T) \geq k$?

MINST-GFI$_f$
INSTANCE: A graph G, and a real number k.
QUESTION: Does G have a spanning tree T with $GFI_f(T) \leq k$?

Our first main contribution is that the above problems are $\mathcal{NP}$-complete if paths are the unique extremal trees in the following sense. Here a path on n vertices, denoted by P_n, is an alternating sequence of distinct vertices and edges $v_1e_1v_2e_2\cdots v_{n-1}e_{n-1}v_n$, such that the vertices v_{i-1} and v_i are the ends of the edge e_{i-1} for all $i \in \{2, 3, \ldots, n\}$.

Theorem 1. *Suppose P_n is the unique tree with the largest value of $GFI_f(T)$ among all spanning trees T of G for every connected graph G on n vertices. Then* MAXST-GFI$_f$ *is $\mathcal{NP}$-complete.*

Theorem 2. *Suppose P_n is the unique tree with the smallest value of $GFI_f(T)$ among all spanning trees T of G for every connected graph G on n vertices. Then* MINST-GFI$_f$ *is $\mathcal{NP}$-complete.*

Since the proofs of both results are similar, we only present the proof of Theorem 1.

Proof. MAXST-GFI$_f$ is clearly in $\mathcal{NP}$, since it is straightforward to check in polynomial time whether $GFI_f(T) \geq k$ for any given spanning tree T of G and real number k. Note that here we implicitly use the assumption that $GFI_f(T)$ can be computed in polynomial time.

To show the $\mathcal{NP}$-completeness of MAXST-GFI$_f$, we use a reduction from the well-known $\mathcal{NP}$-complete problem Hamilton path [13], which is defined as follows.

Hamilton path
INSTANCE: A graph G on n vertices.
QUESTION: Does G have a Hamilton path, i.e., a subgraph isomorphic to P_n?

Suppose that P_n is the unique tree with the largest value of $GFI_f(T)$ among all spanning trees T of G for every connected graph G on n vertices. Since k is not fixed but part of the instances for MAXST-GFI$_f$, we can consider $k = GFI_f(P_n)$. This enables us to show that an arbitrary graph G is a YES-instance of Hamilton path if and only if G is connected and G has a spanning tree T with $GFI_f(T) \geq GFI_f(P_n)$.

Suppose first that G has a Hamilton path. Then G is connected and has a spanning tree T (i.e., the Hamilton path) such that $GIF_f(T) \geq GIF_f(P_n) = k$.

We prove the reverse statement by contradiction. Suppose that G does not have a Hamilton path. Let T be a spanning tree of G such that $GIF_f(T) \geq k = GIF_f(P_n)$. Since P_n is the unique tree with the largest value of $GFI_f(T)$, we have $GIF_f(T) < GIF_f(P_n)$, a contradiction. □

We note here that $\mathcal{NP}$-completeness results similar to the statements in Theorem 1 or Theorem 2 hold for other topological indices, as long as these indices are computable in polynomial time and similar extremal results are known or can be proved. We come back to this in Section 5.

We first continue with some complexity results on $GFI_f(G)$ in case $f(x,y) = \frac{f(x)}{x} + \frac{f(y)}{y}$ for a real function $f(x)$. Recall that this implies $GFI_f(G) = \sum_{u \in V} f(d_G(u))$. We focus on the cases for which we know that $f(x)$ is strictly concave or convex. In these cases we can prove that the above problems MAXST-GFI$_f$ and MINST-GFI$_f$ remain $\mathcal{NP}$-complete when restricted to cubic graphs, i.e., instance graphs in which all vertices have degree 3.

2.2. Conditions on convexity or concavity

In this subsection, throughout we assume $GFI_f(G) = \sum_{u \in V} f(d_G(u))$ for a real function $f(x)$ which is computable in polynomial time. We consider the below special cases of MAXST-GFI$_f$ and MINST-GFI$_f$ for such $GFI_f(G)$ and restricted to cubic graphs. Clearly, if we can show that these special cases c-MINST-GFI$_f$ and c-MAXST-GFI$_f$ are $\mathcal{NP}$-complete, then under the same conditions MINST-GFI$_f$ and MAXST-GFI$_f$ are also $\mathcal{NP}$-complete.

c-MAXST-GFI$_f$
INSTANCE: A cubic graph G, and a real number k.
QUESTION: Does G have a spanning tree T with $GFI_f(T) \geq k$?

c-MINST-GFI$_f$
INSTANCE: A cubic graph G, and a real number k.
QUESTION: Does G have a spanning tree T with $GFI_f(T) \leq k$?

Before presenting our results and proofs, we introduce some additional terminology, notation and a useful lemma. Given a graph $G = (V, E)$, we let $D(G) = [d_1^{a_1}, d_2^{a_2}, \ldots, d_t^{a_t}]$ denote the degree sequence of G, where $d_1 > d_2 > \cdots > d_t$, $a_1 + a_2 + \cdots + a_t = |V|$, and G has exactly a_i vertices with degree d_i. We will frequently use the following concept of majorization [26]. Let $A = [a_1, a_2, \ldots, a_n]$ and $B = [b_1, b_2, \ldots, b_n]$ be non-increasing integer sequences of length n. Then *A majorizes B*, denoted by $A \succeq B$, if

$$\sum_{i=1}^{k} a_i \geq \sum_{i=1}^{k} b_i \text{ for } k = 1, 2, \ldots, n-1, \text{ and } \sum_{i=1}^{n} a_i = \sum_{i=1}^{n} b_i.$$

If at least one of the above inequalities is strict, we use $A \succ B$, and we say that A strictly majorizes B. A function $f(x)$ is strictly convex (resp., concave) on an interval $[a, b]$ if for any two points x_1 and x_2 in $[a, b]$ and any λ with $0 < \lambda < 1$, $f(\lambda x_1 + (1-\lambda)x_2) < \lambda f(x_1) + (1-\lambda) f(x_2)$ (resp., $f(\lambda x_1 + (1-\lambda)x_2) > \lambda f(x_1) + (1-\lambda) f(x_2)$).

The following known result is a key ingredient in our later considerations.

Lemma 1 (*[19]*). *Let G and G' be two graphs with the same number of vertices and edges, and with $D(G) \succeq D(G')$. If $f(x)$ is a strictly concave (resp., convex) function, then $GFI_f(G) \leq GFI_f(G')$ (resp., $GFI_f(G) \geq GFI_f(G')$), with equality holding in the inequality if and only if $D(G) = D(G')$.*

A star on n vertices, denoted by $K_{1,n-1}$, is a tree with $n-1$ vertices of degree 1 and a single vertex of degree $n-1$. If T is any tree on n vertices different from P_n (resp., $K_{1,n-1}$), then clearly $D(T) \succ D(P_n)$ (resp., $D(T) \prec D(K_{1,n-1})$). So, we immediately obtain the following result as a consequence of Lemma 1.

Lemma 2. *Let T be a tree on n vertices. If f is a strictly concave (resp., convex) function, then*

(a) *$GFI_f(T) \leq GFI_f(P_n)$ (resp., $GFI_f(T) \geq GFI_f(P_n)$), with equality holding in the inequality if and only if $T \cong P_n$;*

(b) *$GFI_f(T) \geq GFI_f(K_{1,n-1})$ (resp., $GFI_f(T) \leq GFI_f(K_{1,n-1})$), with equality holding in the inequality if and only if $T \cong K_{1,n-1}$.*

Note that the statements in Lemma 2(a) can be considered as special cases of the statements in Theorems 1 and 2, for the case that $GFI_f(G) = \sum_{u \in V} f(d_G(u))$. The following results show the stronger versions of the latter case when restricted to cubic graphs.

Theorem 3. *If f is a strictly concave function, then* c-MINST-GFI$_f$ *is $\mathcal{NP}$-complete.*

Theorem 4. *If f is a strictly convex function, then* c-MAXST-GFI$_f$ *is $\mathcal{NP}$-complete.*

We show examples of known topological indices satisfying these conditions of results in Section 5. We only prove Theorem 3; the counterpart for strictly convex functions can be proved in a similar way. We show how we can use the following problem 1,3-ST to prove Theorem 3.

1,3-ST INSTANCE: A cubic graph G. QUESTION: Does G have a spanning tree with no vertices of degree 2?

The problem 1,3-ST is known to be $\mathcal{NP}$-complete by a result of Lemke [21].

We complete this section with our proof of Theorem 3.

Proof of Theorem 3. We assume that f is a strictly concave function, and our aim is to prove that the problem c-MINST-GFI$_f$ is $\mathcal{NP}$-complete.

The problem is clearly in $\mathcal{NP}$.

Let T^* be a tree on n vertices with degree sequence $[3^{\frac{n-2}{2}}, 1^{\frac{n+2}{2}}]$ (where n is even by the well-known fact that the degree-sum is twice the number of edges). We consider the problem c-MINST-GFI$_f$ for $k = GFI_f(T^*)$. We prove the required $\mathcal{NP}$-completeness by a reduction from the 1,3-ST problem. Let G be a cubic graph with vertex set $\{v_1, v_2, \ldots, v_n\}$. It is sufficient to prove that G has a spanning tree with no vertices of degree 2 if and only if G has a spanning tree T with $GFI_f(T) \le GFI_f(T^*)$.

Suppose first that T' is a spanning tree of G with no vertices of degree 2. So the degree of each vertex of T' is either 3 or 1. Suppose there are a vertices with degree 3, and b vertices with degree 1. Because T' has n vertices, we have $a+b = n$. Since T' is a spanning tree on n vertices, the number of edges of T' is $n-1$. Since the sum of the degrees is equal to twice the number of edges, we have $3a + b = 2(n-1)$. Thus $a+b = n$ and $3a+b = 2(n-1)$. By straightforward calculations, we obtain $a = \frac{n-2}{2}$ and $b = \frac{n+2}{2}$. So we have $D(T') = [3^{\frac{n-2}{2}}, 1^{\frac{n+2}{2}}] = D(T^*)$. Thus $GFI_f(T') = GFI_f(T^*)$, and hence G has a spanning tree $T = T'$ with $GFI_f(T) \le GFI_f(T^*)$.

For the other implication, suppose that all spanning trees of G have at least one vertex of degree 2. Let T be an arbitrary spanning tree of G. We complete the proof by showing that $GFI_f(T) > GFI_f(T^*)$.

Set $D(T) = [3^{a_1}, 2^{a_2}, 1^{a_3}] = [d_1, d_2, \ldots, d_n]$. So we have $a_2 \ge 1$. Then $a_1 + a_2 + a_3 = n$ and $3a_1 + 2a_2 + a_3 = 2(n-1)$. By straightforward calculations, we have $a_1 = \frac{n-a_2-2}{2}$ and $a_3 = \frac{n-a_2+2}{2}$. Since $a_2 \ge 1$, we have $a_1 < \frac{n-2}{2}$ and $a_3 < \frac{n+2}{2}$. Recall that $D(T^*) = [3^{\frac{n-2}{2}}, 1^{\frac{n+2}{2}}] = [d'_1, d'_2, \ldots, d'_n]$. It is easy to check the validity of the following inequalities.

$$\sum_{i=1}^{k} d_i = 3k = \sum_{i=1}^{k} d'_i \text{ for } k = 1, 2, \ldots, a_1;$$

$$\sum_{i=1}^{k} d_i = 3a_1 + 2(k-a_1) < 3k = \sum_{i=1}^{k} d'_i \text{ for } k = a_1+1, a_1+2, \ldots, \frac{n-2}{2};$$

$$\sum_{i=1}^{k} d_i = 2n-2-2t-a_3 \le 2n-2-t-a_3 = \sum_{i=1}^{k} d'_i$$
$$\text{for } t = n-a_3-k \text{ and } k = \frac{n}{2}, \frac{n}{2}+1, \ldots, a_1+a_2;$$

$$\sum_{i=1}^{k} d_i = 2n-2-\ell = \sum_{i=1}^{k} d'_i$$
$$\text{for } \ell = n-k \text{ and } k = a_1+a_2+1, a_1+a_2+2, \ldots, n.$$

Hence $D(T^*) \succ D(T)$. Since we assume f is strictly concave, using Lemma 1, we conclude that $GFI_f(T) > GFI_f(T^*)$. This completes the proof of Theorem 3. □

3. $\mathcal{APX}$-completeness

In this and the next section, we continue to consider $GFI_f(G) = \sum_{u \in V} f(d_G(u))$ for a real function $f(x)$ which is computable in polynomial time.

However, we will turn our attention to the optimization problems associated with the decision problems we considered in the previous sections.

Adopting the way optimization problems are presented in a classic paper by Johnson [18], we list the optimization versions of c-MINST-GFI$_f$ and c-MAXST-GFI$_f$ as follows.

c-MINST-GFI$_f$
INSTANCE: A cubic graph G.
FEASIBLE SOLUTION: A spanning tree T of G.
OBJECTIVE FUNCTION: $GFI_f(T)$.
OPT: Min.

c-MAXST-GFI$_f$
INSTANCE: A cubic graph G.
FEASIBLE SOLUTION: A spanning tree T of G.
OBJECTIVE FUNCTION: $GFI_f(T)$.
OPT: Max.

The following results deal with the $\mathcal{APX}$-completeness of the above optimization versions of c-MINST-GFI$_f$ and c-MAXST-GFI$_f$. These results imply that there exists an $\epsilon > 0$ such that no polynomial time $(1+\epsilon)$-approximation algorithm is possible for these two problems, unless $\mathcal{P} = \mathcal{NP}$ [2].

Theorem 5. *If f is a strictly concave function, then the optimization version of* c-MINST-GFI$_f$ *is $\mathcal{APX}$-complete.*

Theorem 6. *If f is a strictly convex function, then the optimization version of* c-MAXST-GFI$_f$ *is $\mathcal{APX}$-complete.*

We only prove Theorem 5, since the proof of Theorem 6 is similar. We first give some remarks. The above results directly imply that the optimization version of MINST-GFI$_f$ (resp., MAXST-GFI$_f$) is $\mathcal{APX}$-complete if f is a strictly concave (resp., convex) function and $GFI_f(G) = \sum_{u\in V} f(d_G(u))$ for a real function $f(x)$ which is computable in polynomial time.

There exists an extensive literature on proofs for $\mathcal{APX}$-completeness of optimization problems by L-reductions (see, e.g., [1,8,27,28]). Given two optimization problems F and G, and a polynomial time transformation h from instances of F to instances of G, we say that h is an *L-reduction* if there are positive constants α and β such that for every instance x of F,

1. $\mathrm{opt}_G(h(x)) \le \alpha \cdot \mathrm{opt}_F(x)$;
2. for every feasible solution y of $h(x)$ with objective value $g_G(h(x), y) = c_2$, we can in polynomial time find a solution y' of x with $g_F(x, y') = c_1$ such that $|\mathrm{opt}_F(x) - c_1| \le \beta \cdot |\mathrm{opt}_G(h(x)) - c_2|$.

We next prove Theorem 5 by an L-reduction; the counterpart for strictly convex functions can be proved in a similar way. For the full proof of Theorem 5 we also need to show that c-MINST-GFI$_f \in \mathcal{APX}$. This will be done in Section 4.

Proof of Theorem 5. Next we prove the $\mathcal{APX}$-hardness by an L-reduction from the optimization problem c-MLST to c-MINST-GFI$_f$. The c-MLST problem is defined as follows; it is known to be $\mathcal{APX}$-complete by a result due to Bonsma [4].

c-MLST
INSTANCE: A cubic graph G.
FEASIBLE SOLUTION: A spanning tree T of G.
OBJECTIVE FUNCTION: The number of leaves of T.
OPT: Max.

Let $G = (V, E)$ be a cubic graph on n vertices. Let T_1 and T_2 be two spanning trees of G on distinct numbers of leaves ℓ_1 and ℓ_2, respectively. It is easy to see that $D(T_i) = [3^{\ell_i-2}, 2^{n+2-2\ell_i}, 1^{\ell_i}]$ for $i = 1, 2$. Under the majorization relation, any pair of degree sequences of two spanning trees of G are comparable. This implies either $D(T_1) \succ D(T_2)$ or $D(T_1) \prec D(T_2)$. By straightforward calculations, $\ell_1 > \ell_2$ if and only if $D(T_1) \succ D(T_2)$. By Lemma 1, we have $GFI_f(T_1) < GFI_f(T_2)$ if and only if $\ell_1 > \ell_2$. This implies that a spanning tree T^* of G has the maximum number ℓ^* of leaves if and only if $GFI_f(T^*)$ attains the minimum value among all spanning trees of G. Let T be a spanning tree of G with $GFI_f(T) = (\ell-2)f(3) + (n+2-2\ell)f(2) + \ell f(1)$. This implies that T has ℓ leaves. Since $f(x)$ is strictly concave, we have

$$
\begin{aligned}
&\left|\frac{(\ell^*-2)f(3)+(n+2-2\ell^*)f(2)+\ell^* f(1)}{\ell^*}\right| \\
&\le \left|\frac{\ell^*(f(3)-2f(2)+f(1))}{\ell^*}\right| + \left|\frac{(n+2)f(2)}{\ell^*}\right| + \left|\frac{2f(3)}{\ell^*}\right| \\
&\le 2f(2)-f(3)-f(1)+\frac{(n+2)|f(2)|}{2}+|f(3)| \\
&\le 2|f(3)|+\frac{(n+6)|f(2)|}{2}-f(1)
\end{aligned}
$$

and

$$\left|\frac{\ell^* - \ell}{(\ell^* - \ell)f(3) - 2(\ell^* - \ell)f(2) + (\ell^* - \ell)f(1)}\right| = \frac{1}{2f(2) - f(3) - f(1)}.$$

Therefore,

$$\left|(\ell^* - 2)f(3) + (n + 2 - 2\ell^*)f(2) + \ell^* f(1)\right| \le \alpha \cdot |\ell^*|$$

and

$$|\ell^* - \ell| \le \beta \cdot \left|(\ell^* - \ell)f(3) - 2(\ell^* - \ell)f(2) + (\ell^* - \ell)f(1)\right|,$$

where $\alpha = 2|f(3)| + \frac{(n+6)|f(2)|}{2} - f(1)$ and $\beta = \frac{1}{2f(2)-f(3)-f(1)}$. □

4. Approximation algorithm

In this section, we prove the following results for the optimization problems we introduced in the previous section. As before, we only give the details for the minimization version.

Theorem 7. *If f is a strictly concave function, then* c-MINST-GFI$_f$ $\in \mathcal{APX}$.

Theorem 8. *If f is a strictly convex function, then* c-MAXST-GFI$_f$ $\in \mathcal{APX}$.

We prove Theorem 7 by showing that the next algorithm is a polynomial-time approximation algorithm for the optimization version of c-MINST-GFI$_f$. An explanation of the notation and rationale of the algorithm follows.

Algorithm 1: LOCAL SEARCH-c-MINST-GFI$_f$

Input: a cubic graph G.
Output: a spanning tree T' and $GFI_f(T')$.
Initialize a spanning tree T of G;
$\widehat{E} \leftarrow E(G) \setminus E(T)$;
while $\widehat{E} \neq \emptyset$ **do**
  randomly choose an edge $e \in \widehat{E}$;
  if $e \in E^*(T+e)$ **then**
    $\widehat{E} \leftarrow \widehat{E} \setminus \{e\}$;
  else
    randomly choose an edge $e^* \in E^*(T+e)$;
    $T \leftarrow T + e - e^*$;
    $\widehat{E} \leftarrow E(G) \setminus E(T)$;
  end
end
Return $T' \leftarrow T$ and $GFI_f(T') \leftarrow GFI_f(T)$.

Let $G = (V, E)$ be a graph. Define $h_G(w) = f(d_G(w)) - f(d_G(w) - 1)$ for $w \in V(G)$. Suppose that $e \in E(G)$ is an edge joining vertices $u \in V(G)$ and $v \in V(G)$. Define $\Delta_G(e) = h_G(u) + h_G(v)$. Let $\Delta^*_G = \max_{e \in E(G)}\{\Delta_G(e) \mid G - e \text{ is connected}\}$ and $E^*(G) = \{e \mid \Delta_G(e) = \Delta^*_G \text{ and } G - e \text{ is connected}\}$. Our local search algorithm is based on the following observation: if $e^* \in E^*(G)$, then $GFI_f(G - e^*)$ attains the minimum value among all connected spanning subgraphs with $|E(G)| - 1$ edges. If we add a new edge e to a tree T, then we obtain a unicyclic graph. Deleting an edge $e^* \in E^*(T + e)$ gives us a new tree $T + e - e^*$ with $GFI_f(T + e - e^*) \le GFI_f(T)$. This is the rationale behind the algorithm LOCAL SEARCH-c-MINST-GFI$_f$.

In what follows, we first give an example of the execution of the algorithm, referring to Fig. 2 and supposing that $f(x) = \sqrt{x}$.

- Input: a cubic graph G and an initial spanning tree T of G.
- $\hat{E} = E(G) \setminus E(T) = \{e_1, e_5, e_7, e_8\}$ (i.e., the set of deleted edges).
- Choose $e_8 \in \hat{E}$. Calculate $\Delta_{T+e_8}(e_3) = \sqrt{3} - \sqrt{2} + \sqrt{3} - \sqrt{2} \approx 0.635$, $\Delta_{T+e_8}(e_2) = \Delta_{T+e_8}(e_8) = \sqrt{3} - \sqrt{2} + \sqrt{2} - \sqrt{1} \approx 0.732$. Thus $E^*(T + e_8) = \{e_2, e_8\}$.
- $T \to T$ ($e_8 \in E^*(T + e_8)$) and $\hat{E} \setminus \{e_8\} = \{e_1, e_5, e_7\} \to \hat{E}$ ($\hat{E} \neq \emptyset$).

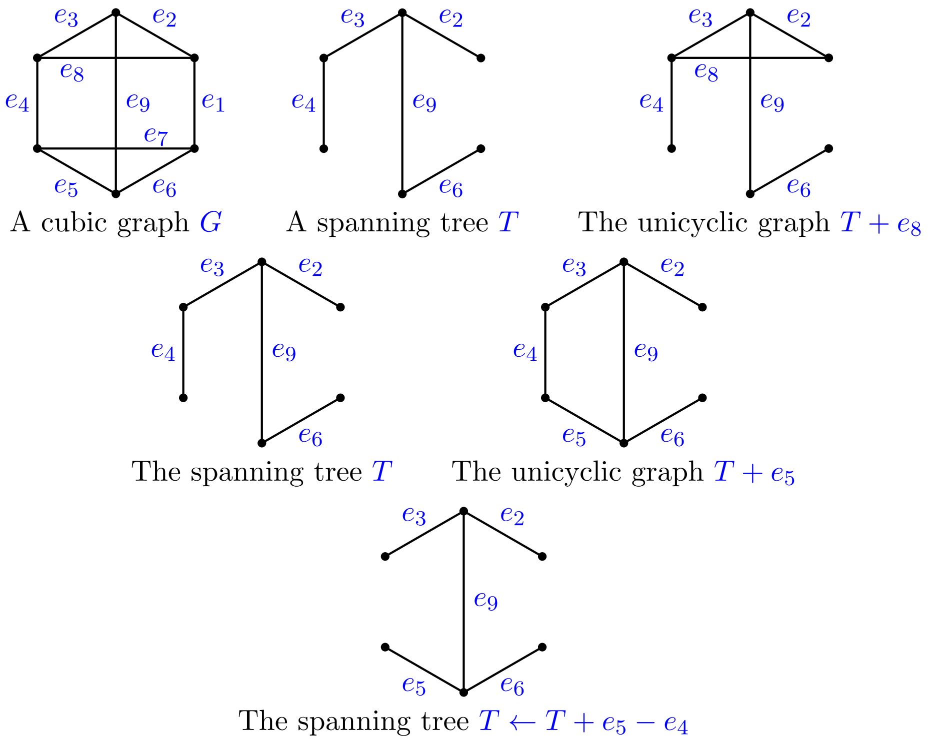

Fig. 2. An example of the algorithm execution.

- Choose $e_5 \in \hat{E} = \{e_1, e_5, e_7\}$. Calculate $\Delta_{T+e_5}(e_3) = \Delta_{T+e_5}(e_5) = \sqrt{3} - \sqrt{2} + \sqrt{2} - \sqrt{1} \approx 0.732$, $\Delta_{T+e_5}(e_4) = \sqrt{2} - \sqrt{1} + \sqrt{2} - \sqrt{1} \approx 0.828$, $\Delta_{T+e_5}(e_9) = \sqrt{3} - \sqrt{2} + \sqrt{3} - \sqrt{2} \approx 0.635$. Thus $E^*(T + e_5) = \{e_4\}$.
- $T + e_5 - e_4 \to T$ ($e_5 \notin E^*(T + e_5)$) and $E(G) \setminus E(T) = \{e_1, e_4, e_7, e_8\} \to \hat{E}$ ($\hat{E} \neq \emptyset$).
- Choose $e_4 \in \hat{E}$. Calculate $\Delta_{T+e_4}(e_4) = \sqrt{2}-\sqrt{1}+\sqrt{2}-\sqrt{1} \approx 0.828$, $\Delta_{T+e_4}(e_3) = \Delta_{T+e_4}(e_5) = \sqrt{3}-\sqrt{2}+\sqrt{2}-\sqrt{1} \approx 0.732$, $\Delta_{T+e_4}(e_9) = \sqrt{3} - \sqrt{2} + \sqrt{3} - \sqrt{2} \approx 0.635$. Thus $E^*(T + e_4) = \{e_4\}$.
- $T \to T$ ($e_4 \in E^*(T + e_4)$) and $\hat{E} \setminus \{e_4\} = \{e_1, e_7, e_8\} \to \hat{E}$ ($\hat{E} \neq \emptyset$).
- Choose $e_1 \in \hat{E}$. Calculate $\Delta_{T+e_1}(e_1) = \sqrt{2}-\sqrt{1}+\sqrt{2}-\sqrt{1} \approx 0.828$, $\Delta_{T+e_1}(e_2) = \Delta_{T+e_1}(e_6) = \sqrt{3}-\sqrt{2}+\sqrt{2}-\sqrt{1} \approx 0.732$, $\Delta_{T+e_1}(e_9) = \sqrt{3} - \sqrt{2} + \sqrt{3} - \sqrt{2} \approx 0.635$. Thus $E^*(T + e_1) = \{e_1\}$.
- $T \to T$ ($e_1 \in E^*(T + e_1)$) and $\hat{E} \setminus \{e_1\} = \{e_7, e_8\} \to \hat{E}$ ($\hat{E} \neq \emptyset$).
- Choose $e_8 \in \hat{E}$. Calculate $\Delta_{T+e_8}(e_8) = \sqrt{2}-\sqrt{1}+\sqrt{2}-\sqrt{1} \approx 0.828$, $\Delta_{T+e_8}(e_2) = \Delta_{T+e_8}(e_3) = \sqrt{3}-\sqrt{2}+\sqrt{2}-\sqrt{1} \approx 0.732$. Thus $E^*(T + e_8) = \{e_8\}$.
- $T \to T$ ($e_8 \in E^*(T + e_8)$) and $\hat{E} \setminus \{e_8\} = \{e_7\} \to \hat{E}$ ($\hat{E} \neq \emptyset$).
- Choose $e_7 \in \hat{E}$. Calculate $\Delta_{T+e_7}(e_7) = \sqrt{2}-\sqrt{1}+\sqrt{2}-\sqrt{1} \approx 0.828$, $\Delta_{T+e_7}(e_5) = \Delta_{T+e_7}(e_6) = \sqrt{3}-\sqrt{2}+\sqrt{2}-\sqrt{1} \approx 0.732$. Thus $E^*(T + e_7) = \{e_7\}$.
- $T \to T$ ($e_7 \in E^*(T + e_7)$) and $\hat{E} \setminus \{e_7\} = \emptyset \to \hat{E}$.
- $T \to T'$ and $GIF_f(T) = 2\sqrt{3} + 4 \to GIF_f(T')$.

Before proving Theorem 7, we take on the job of analysing its running time on an input cubic graph G on n vertices. We may call the breadth-first search algorithm of [7] to generate a spanning tree of G in running time $O(n)$. Let T be a spanning tree of G. Since G is a cubic graph, for any edge $e \in E(G) \setminus E(T)$, Δ^*_{T+e} is one value in the set $\{2f(3)-2f(2), f(3)-f(1), 2f(2)-2f(1)\}$. This implies that, for $e^* \in E^*(T+e)$, $GIF_f(T)-GIF_f(T+e-e^*) \geq 2f(2)-f(3)-f(1)$ if $GIF_f(T+e-e^*) < GIF_f(T)$. By Lemma 2, we have $GIF_f(T) \leq (n-2)f(2)+2f(1)$. The number of iterations of the while-loop is at most $\frac{(n-2)f(2)+2f(1)}{2f(2)-f(3)-f(1)}$, which is $O(n)$. The running time of each iteration corresponds to the worst case of times we need to optimize a spanning tree T. Since G is a cubic graph on n vertices, $|E(G)| = \frac{3n}{2}$. The worst case (all edges are local optima) is that we need to consider each edge in $\hat{E} = E(G) \setminus E(T)$ (that is, consider all unicyclic graphs generated by adding one edge). We have $|\hat{E}| = |E(G)| - (n-1) = \frac{n}{2} + 1$. For each unicyclic graph, we need to calculate which edge in the cycle is optimal. The worst case is that all edges we need to consider are in the cycle. In that case we need to calculate this n times. Each iteration runs in time $(\frac{n}{2} + 1) \times n = O(n^2)$. Therefore, Local Search-c-MINST-GFI$_f$ runs in time $O(n^3)$, where n is the number of vertices of the input cubic graph G.

Let G be a graph. A vertex v of G is called a *pendant vertex* if and only if v has degree 1. Now we have all the ingredients to prove Theorem 7.

Proof of Theorem 7. Let G be a cubic graph on n vertices. Let T' be a spanning tree of G obtained by using Algorithm 1. Let a, b and c be the numbers of vertices of degree 3, 2 and 1 of T', respectively. We first state and prove two claims.

Claim 1. $a < c$.

Proof. Using the above notations, we have $3a + c = 2n - 2b - 2$ and $a + c = n - b$. This implies $a - c = -2 < 0$. □

An M-path P is a maximal path with all internal vertices of degree 2 and ends of degrees 1 or 3. It follows that the sum of the numbers of internal vertices of M-paths of T' is b.

Claim 2. $b \le 4c$.

Proof. By Claim 1, the number of M-paths is less than $2c$. If each M-path has at most two internal vertices, then the claim holds. If all internal vertices of every M-path are adjacent to pendant vertices of T', then the claim holds as well. This follows since each pendant vertex of T' is adjacent to at most two internal vertices, hence $b \le 2c \le 4c$.

Let P be an M-path. Suppose that there exist at least three internal vertices of P, with one internal vertex not adjacent to any pendant vertex of T'.

Suppose that u is an internal vertex of P and w is a vertex in G with $d_{T'}(w) = 2$ satisfying $e = uw \in E(G) \setminus E(T')$.

Set $\{v, x\} = N_P(u)$. Since T is a tree, $e = wu$ must lie on the cycle of $T' + e$. There exists a path, different from wu, from w to u passing through v or x. This implies that v or x is on the cycle of $T' + e$. Otherwise, $d_T(u) \ge 3$ is not an internal vertex of P. Without loss of generality, we assume that v is on the cycle of $T' + e$. Set $e' = uv \in E(P)$. It follows that $T' + e - e'$ is a spanning tree of G. By straightforward calculations, we obtain the following expression for $\Delta_{T'+e}(e') - \Delta_{T'+e}(e)$:

$$\begin{aligned}
&\Delta_{T'+e}(e') - \Delta_{T'+e}(e)\\
=&(f(d_{T'+e}(u)) - f(d_{T'+e}(u) - 1) + f(d_{T'+e}(v)) - f(d_{T'+e}(v) - 1))\\
&- (f(d_{T'+e}(u)) - f(d_{T'+e}(u) - 1) + f(d_{T'+e}(w)) - f(d_{T'+e}(w) - 1))\\
=&f(d_{T'+e}(v)) - f(d_{T'+e}(v) - 1) - f(d_{T'+e}(w)) + f(d_{T'+e}(w) - 1)\\
=&f(2) - f(1) - f(3) + f(2)\\
=&2f(2) - f(3) - f(1).
\end{aligned}$$

Recall that a real-valued function f on an interval is said to be strictly concave if $f((1-\alpha)x + \alpha y) > (1-\alpha)f(x) + \alpha f(y)$ for any $\alpha \in (0, 1)$ and $x \ne y$. Setting $x = 3$, $y = 1$ and $\alpha = \frac{1}{2}$, we get $f(2) > \frac{1}{2}f(3) + \frac{1}{2}f(1)$ (i.e., $2f(2) - f(3) - f(1) > 0$). So we obtain $\Delta_{T'+e}(e') > \Delta_{T'+e}(e)$. This implies $GFI_f(T' + e - e') < GFI_f(T')$, which contradicts that T' is a minimal spanning tree of G constructed by Algorithm 1.

This completes the proof of Claim 2. □

Let $\widehat{T}$ be a spanning tree of G with the minimum graphical function-index. Let T^* be a tree with degree sequence $D(T^*) = [3^{\frac{n-2}{2}}, 1^{\frac{n+2}{2}}]$. From the proof of Theorem 3, we have $GFI_f(T^*) \le GFI_f(\widehat{T})$ and $c \le \frac{n+2}{2}$. By Claims 1 and 2, we have $n = a + b + c < 6c$. This implies $a + b < \frac{5n}{6}$. Since G is a cubic graph, we have $n \ge 4$. If $f(3) \ge f(2)$, then we get

$$\begin{aligned}
GFI_f(G)(T') &= af(3) + bf(2) + cf(1)\\
&\le (a+b)f(3) + cf(1)\\
&\le \frac{5n}{6}f(3) + \frac{n+2}{2}f(1)\\
&\le \frac{10}{3}\left(\frac{n-2}{2}f(3) + \frac{n+2}{2}f(1)\right)\\
&= \frac{10}{3}GFI_f(T^*)\\
&\le \frac{10}{3}GFI_f(\widehat{T}).
\end{aligned}$$

If $f(3) < f(2)$, then we get

$$\begin{aligned}
GFI_f(G)(T') &= af(3) + bf(2) + cf(1)\\
&\le \frac{5n}{6}f(2) + \frac{n+2}{2}f(1)
\end{aligned}$$

Table 2
Some topological indices and their definition.

Name	Expression
Balaban index	$J(G)=\frac{\vert E(G)\vert}{\mu+1}\sum_{uv\in E(G)}\frac{1}{\sqrt{\delta_G(u)\delta_G(v)}}$
Degree-based entropy	$I_k(G)=-\sum_{u\in V(G)}\frac{d_G(u)^k}{\sum_{v\in V(G)}d_G(v)^k}\log_2\left(\frac{d_G(u)^k}{\sum_{v\in V(G)}d_G(v)^k}\right)$
First degree-based entropy	$I_1(G)=-\sum_{u\in V(G)}\frac{d_G(u)}{2\vert E(G)\vert}\log_2\left(\frac{d_G(u)}{2\vert E(G)\vert}\right)$
Energy	$EN(G)=\sum_{i=1}^{n}\vert\lambda_i\vert$
Estrada index	$EE(G)=\sum_{i=1}^{n}e^{\lambda_i}$
First Zagreb index	$M_1(G)=\sum_{v\in V(G)}d_G(v)^2$
General first Zagreb index	$M_1^{\alpha}(G)=\sum_{v\in V(G)}d_G(v)^{\alpha}$
General Randić index	$R_{\alpha}(G)=\sum_{uv\in E(G)}(d_G(u)d_G(v))^{\alpha}$
Geometric-arithmetic index	$GA(G)=\sum_{uv\in E(G)}\frac{2\sqrt{d_G(u)d_G(v)}}{d_G(u)+d_G(v)}$
Harmonic index	$H(G)=\sum_{uv\in E(G)}\frac{2}{d_G(u)+d_G(v)}$
Hosoya index	$Z(G)=\sum_{k\geq 0}m(G,k)$
Hyper-Wiener index	$WW(G)=\frac{1}{2}\left(\sum_{u,v\in V(G)}d_G(u,v)+\sum_{u,v\in V(G)}d_G^2(u,v)\right)$
Merrifield–Simmons index	$\sigma(G)=\sum_{k\geq 0}i(G,k)$
Randić index	$R(G)=\sum_{uv\in E(G)}\frac{1}{\sqrt{d_G(u)d_G(v)}}$
Second Zagreb index	$M_2(G)=\sum_{uv\in E(G)}d_G(u)d_G(v)$
Sum-connectivity index	$\chi(G)=\sum_{uv\in E(G)}\frac{1}{\sqrt{d_G(u)d_G(v)}}$
Wiener index	$W(G)=\sum_{\{u,v\}\subseteq V(G)}d_G(u,v)$

$$
\begin{aligned}
&\leq \frac{10f(2)}{3f(3)}\left(\frac{n-2}{2}f(3)+\frac{n+2}{2}f(1)\right)\\
&=\frac{10f(2)}{3f(3)}GFI_f(T^*)\\
&\leq \frac{10f(2)}{3f(3)}GFI_f(\widehat{T}).
\end{aligned}
$$

It follows that $GFI_f(G)(T')\leq\gamma\cdot GFI_f(\widehat{T})$ in which $\gamma=\max\{\frac{10}{3},\frac{10f(2)}{3f(3)}\}$. □

5. Concluding remarks

To complete the paper, we reflect on an earlier remark by listing some examples of well-studied topological indices in Table 2 that do not fall under our general description, but for which similar complexity results hold. Following the table, we gathered the known associated extremal results from literature in Theorem 9. The consequences of these extremal results for the complexity of our studied decision and optimization problems are summarized in two corollaries.

For a better understanding of the expressions of the topological indices that are listed in the table, we introduce some additional notation, without going into the details.

Let G be a graph. As usual, with $d_G(u,v)$ we denote the distance between two vertices u and v in G, assuming that G is connected. Under the same assumption, we use $\delta_G(u)$ to denote the distance sum of a vertex $u\in V(G)$ to all other vertices in G. In the table, $\mu=|E(G)|-|V(G)|+1$ is used for the cyclomatic number, and $\lambda_1,\lambda_2,\ldots,\lambda_{|V(G)|}$ indicate the eigenvalues of the adjacency matrix of G. Furthermore, α is used to indicate a real number, k is an integer, and $m(G,k)$ denotes the number of distinct matchings of G consisting of k edges. So, by definition, $m(G,0)=1$ and $m(G,1)=|E(G)|$. Finally, in the table $i(G,k)$ denotes the number of distinct k-element independent vertex sets of G. So, by definition, $i(G,0)=1$ and $i(G,1)=|V(G)|$.

Below we present the known extremal results with respect to the listed topological indices, together with their sources.

Theorem 9. *Among trees on n vertices, P_n is the unique extremal tree with the minimum Balaban index [10,11], the maximum degree-based graph entropy for $k > 0$ [17], the maximum energy [23], the minimum Estrada index [9], the minimum general first Zagreb index $M_1^\alpha(G)$ for $\alpha < 0$ or $\alpha > 1$ [24], the maximum general first Zagreb index $M_1^\alpha(G)$ for $0 < \alpha < 1$ [24], the minimum general Randić index $R_\alpha(G)$ for $0 < \alpha \leq 1$ [16], the maximum general Randić $R_\alpha(G)$ for $\alpha < 0$ [15], the maximum geometrical–arithmetic index [31], the maximum harmonic index for $n \geq 4$ [34], the maximum Hosoya index [32], the maximum hyper-Wiener index [14], the minimum Merrifield–Simmons index [32], the minimum second Zagreb index [20], the maximum sum-connectivity index [5], and the maximum Wiener index [12].*

The above extremal results have the following consequences. In the first corollary, the decision version of the maximization problem for a specific topological index $TI(G)$ is the problem of deciding whether G has a spanning tree T with $TI(T) \geq k$, for an arbitrary graph G and real number k. The decision versions of the minimization problems are defined analogously.

Corollary 1. *The decision version of the maximization problem is $\mathcal{NP}$-complete for the following topological indices: the degree-based entropy for $k > 0$, the energy, the general first Zagreb index $M_1^\alpha(G)$ for $0 < \alpha \leq 1$, the general Randić index R_α for $\alpha < 0$, the harmonic index, the Hosoya index, the hyper-Wiener index, the sum-connectivity index, and the Wiener index.*

The decision version of the minimization problem is $\mathcal{NP}$-complete for the following topological indices: the Balaban index, the Estrada index, the general first Zagreb index $Z_1^\alpha(G)$ for $\alpha < 0$ or $\alpha > 1$, the general Randić index $R_\alpha(G)$ for $0 < \alpha \leq 1$, the Merrifield–Simmons index, and the second Zagreb index.

The general first Zagreb index M_1^α (resp., the first degree-based entropy I_1) corresponds to $f(x) = x^\alpha$ (resp., $f(x) = \frac{x}{2|E(G)|}\log_2(\frac{2|E(G)|}{x})$). We immediately obtain that $f(x)$ regarding M_1^α is strictly concave for $1 < \alpha < 1$, and strictly convex for $\alpha < 0$ or $\alpha > 1$; $f(x)$ regarding I_1 is strictly concave. By applying Theorems 3, 4, 5 and 6, we obtain the following result.

Corollary 2. *The maximization problem is $\mathcal{NP}$-complete and $\mathcal{APX}$-complete for the general first Zagreb index M_1^α for $\alpha < 0$ or $\alpha > 1$.*

The minimization problem is $\mathcal{NP}$-complete and $\mathcal{APX}$-complete for the following indices: the general first Zagreb index M_1^α for $0 < \alpha < 1$, and the first degree-based entropy I_1.

Data availability

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

Acknowledgements

The authors are very grateful to the referees for their comments and suggestions, which improved the presentation.

References

[1] P. Alimonti, V. Kann, Some APX-completeness results for cubic graphs, Theoret. Comput. Sci. 237 (2000) 123–134.
[2] S. Arora, C. Lund, R. Motwani, M. Sudan, M. Szegedy, Proof verification and the hardness of approximation problems, J. AMC 45 (1998) 501–555.
[3] J.A. Bondy, U.S.R. Murty, Graph Theory, Springer, New York, 2008.
[4] P. Bonsma, Max-leaves spanning tree is APX-hard for cubic graphs, J. Discrete Algorithms 12 (2012) 14–23.
[5] G. Caporossi, I. Gutman, P. Hansen, L. Pavlović, Graphs with maximum connectivity index, Comput. Biol. Chem. 27 (2003) 85–90.
[6] D. Cheng, Q. Zhu, J. Huang, Q. Wu, L. Yang, Clustering with local density peaks-based minimum spanning tree, IEEE Trans. Knowl. Data Eng. 33 (2021) 2930056.
[7] T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein, Introduction to Algorithms, MIT Press, Cambridge, 2009.
[8] P. Crescenzi, A short guide to approximation preserving reductions, in: Twelfth Annual IEEE Conference on Computational Complexity, Ulm, 1997, IEEE Computer Soc., Los Alamitos, CA, 1997, pp. 262–273.
[9] K. Das, S.-G. Lee, On the Estrada index conjecture, Linear Algebra Appl. 431 (2009) 1351–1359.
[10] H. Deng, On the Balaban index of trees, MATCH Commun. Math. Comput. Chem. 66 (2011) 253–260.
[11] H. Dong, X. Guo, Character of graphs with extremal Balaban index, MATCH Commun. Math. Comput. Chem. 63 (2010) 799–812.
[12] R.C. Entringer, D.E. Jackson, D.A. Snyder, Distance in graphs, Czechoslovak Math. J. 26 (1976) 283–296.
[13] M.R. Garey, D.S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, in: A Series of Books in the Mathematical Sciences, W.H. Freeman and Co., San Francisco, 1979.
[14] I. Gutman, W. Linert, I. Lukovits, A.A. Dobrynin, Trees with extremal hyper-Wiener index: Mathematical basis and chemical applications, J. Chem. Inf. Comput. Sci. 37 (1997) 349–354.
[15] Y. Hu, X. Li, Y. Yuan, Trees with maximum general Randić index, MATCH Commun. Math. Comput. Chem. 52 (2004) 129–146.
[16] Y. Hu, X. Li, Y. Yuan, Trees with minimum general Randić index, MATCH Commun. Math. Comput. Chem. 52 (2004) 119–128.
[17] A. Ilić, On the extremal values of general degree-based graph entropies, Inform. Sci. 370–371 (2016) 424–427.
[18] D.S. Johnson, Approximation algorithms for combinatorial problems, J. Comput. System Sci. 9 (1974) 256–278.
[19] J. Karamata, Sur une inégalité rélative aux fonctions convexes, Publ. Math. Univ. Belgrade 9 (1932) 145–148.
[20] R. Lang, X. Li, S. Zhang, Inverse problem for the zagreb index of molecular graphs, Appl. Math. J. Chinese Univ. Ser. A 18 (2003) 487–493.
[21] P. Lemke, The Maximum-Leaf Spanning Tree Problem in Cubic Graphs Is NP-Complete, in: IMA Preprinter Series #428, University of Minnesota, Minneapolis, 1988.
[22] X. Li, D. Peng, Extremal problems for graphical function-indices and f-weighted adjacency matrix, Discrete Math. Lett. 9 (2022) 57–66.

[23] X. Li, Y. Shi, I. Gutman, Graph Energy, Springer, New York, 2012.
[24] X. Li, J. Zheng, A unified approach to the extremal trees for different indices, MATCH Commun. Math. Comput. Chem. 50 (2005) 195–208.
[25] X. Liu, W. Fa, A fully automatic algorithm for reflector detection in radargrams based on continuous wavelet transform and minimum spanning tree, IEEE Trans. Geosci. Remote Sens. 60 (2022) 4601620.
[26] A.W. Marshall, I. Olkin, B.C. Arnold, Inequalities: Theory of Majorization and Its Applications, Springer, New York, 1976.
[27] C.H. Papadimitriou, M. Yannakakis, Optimization, approximation, and complexity classes, J. Comput. System Sci. 43 (1991) 425–440.
[28] A. Reich, Complexity of the maximum leaf spanning tree problem on planar and regular graphs, Theoret. Comput. Sci. 626 (2016) 134–143.
[29] N. Speer, P. Men, C. Spieth, A. Zell, Clustering gene expression data with memetic algorithms based on minimum spanning trees, in: The 2003 Congress on Evolutionary Computation, IEEE, Canberra, 2003, pp. 1848–1855.
[30] I. Tomescu, Extremal vertex-degree function index for trees and unicyclic graphs with given independence number, Discrete Appl. Math. 306 (2022) 83–88.
[31] D. Vukičević, B. Furtula, Topological index based on the ratios of geometrical and arithmetical means of end-vertex degrees of edges, J. Math. Chem. 46 (2009) 1369–1376.
[32] S. Wagner, I. Gutman, Maxima and minima of the Hosoya index and the Merrifield–Simmons index: A survey of results and techniques, Acta Appl. Math. 112 (2010) 323–346.
[33] Y. Yao, M. Liu, F. Belardo, C. Yang, Unified extremal results of topological indices and spectral invariants of graphs, Discrete Appl. Math. 271 (2019) 218–232.
[34] L. Zhong, The harmonic index for graphs, Appl. Math. Lett. 25 (2013) 562–566.