Fractals for Kernelization Lower Bounds, With an Application to Length-Bounded Cut Problems*

Till Fluschnik[†1], Danny Hermelin[‡2], André Nichterlein[§3], and Rolf Niedermeier[4]

1 **Institut für Softwaretechnik und Theoretische Informatik, TU Berlin, Berlin, Germany**
`till.fluschnik@tu-berlin.de`

2 **Department of Industrial Engineering and Management, Ben-Gurion University of the Negev, Israel**
`hermelin@bgu.ac.il`

3 **Institut für Softwaretechnik und Theoretische Informatik, TU Berlin, Berlin, Germany**
`andre.nichterlein@tu-berlin.de`

4 **Institut für Softwaretechnik und Theoretische Informatik, TU Berlin, Berlin, Germany**
`rolf.niedermeier@tu-berlin.de`

Abstract

Bodlaender et al.'s [6] cross-composition technique is a popular method for excluding polynomial-size problem kernels for NP-hard parameterized problems. We present a new technique exploiting triangle-based fractal structures for extending the range of applicability of cross-compositions. Our technique makes it possible to prove new no-polynomial-kernel results for a number of problems dealing with length-bounded cuts. Roughly speaking, our new technique combines the advantages of serial and parallel composition. In particular, answering an open question of Golovach and Thilikos [13], we show that, unless NP $\subseteq$ coNP / poly, the NP-hard Length-Bounded Edge-Cut problem (delete at most k edges such that the resulting graph has no s-t path of length shorter than ℓ) parameterized by the combination of k and ℓ has no polynomial-size problem kernel. Our framework applies to planar as well as directed variants of the basic problems and also applies to both edge and vertex deletion problems.

1998 ACM Subject Classification F.2.2 Nonnumerical Algorithms and Problems, G.2.2 [Discrete Mathematics] Graph Theory

Keywords and phrases Parameterized complexity, polynomial-time data reduction, cross-compositions, lower bounds, graph modification problems, interdiction problems

Digital Object Identifier 10.4230/LIPIcs.ICALP.2016.25

* A full version of the paper is available at `http://arxiv.org/abs/1512.00333`.

† Till Fluschnik acknowledges support by the DFG, project DAMM (NI 369/13-2).

‡ Danny Hermelin was supported by a DFG Mercator fellowship within the project DAMM (NI 369/13-2) while staying at TU Berlin (August 2015). He has also received funding from the People Programme (Marie Curie Actions) of the European Union's Seventh Framework Programme (FP7/2007-2013) under REA grant agreement number 631163.11, and by the ISRAEL SCIENCE FOUNDATION (grant No. 551145/14).

§ André Nichterlein is from February 2016 to January 2017 on postdoctoral leave to Durham University (GB), funded by DAAD.

1 Introduction

Lower bounds are of central concern all over computational complexity analysis. With respect to fixed-parameter tractable problems, currently there are two main streams in this context: (i) ETH-based lower bounds for the running times of exact algorithms [18] and (ii) lower bounds on problem kernel sizes; more specifically, the exclusion of polynomial-size problem kernels [17]. Both these research directions for lower bounds rely on plausible complexity-theoretic assumptions, namely the Exponential-Time Hypothesis (ETH) and NP $\not\subseteq$ coNP / poly, respectively. In this work, we contribute to the second research direction, developing a new technique that exploits a triangle-based fractal structure in order to exclude polynomial-size problem kernels (polynomial kernels for short) for edge and vertex deletion problems in the context of length-bounded cuts.

Kernelization is a key method for designing fixed-parameter algorithms [14, 17]; among all techniques of parameterized algorithm design, it has the presumably greatest potential for delivering practically relevant algorithms. Hence, it is of key interest to explore its power and its limitations. In a nutshell, the fundamental idea of kernelization is as follows. Given a parameterized problem instance I with parameter k, in polynomial time preprocess I by applying data reduction rules in order to simplify it and reduce it to an "equivalent" instance (so-called (problem) kernel) of the same problem. For NP-hard problems the best one can hope for is a problem kernel of size polynomial in the parameter k. In a way, one may interpret kernelization (requested to run in polynomial time) as an "exact counterpart" of polynomial-time approximation algorithms. Indeed, linear-size problem kernels often imply constant-factor approximation algorithms [20, page 15]. Approximation algorithmics has a highly developed theory (having produced concepts such as MaxSNP-hardness and the famous PCP theory) for proving (relative to some plausible complexity-theoretic assumption) lower bounds on the approximation factors [24].

It is fair to say that in the younger field of kernelization the arsenal for proving lower bounds (particularly excluding polynomial kernels) so far is of smaller scope and needs further development. The most influential result in this context is due to Bodlaender et al. [5] and Fortnow and Santhanam [12]: Based on the assumption NP $\not\subseteq$ coNP / poly, it is shown that e.g. the NP-hard graph problem Longest Path parameterized by solution size has no polynomial kernel. The core tool for showing this are so-called "OR-compositions". To ease the use of this kernel-lower-bound framework, one natural idea is to use "polynomial parameter transformations", that is, a form of "parameter-preserving reductions" [7, 10]. An easier-to-use generalization of the OR-composition technique is given by so-called OR-cross-compositions [6]. Currently, these two approaches constitute the known core tools to exclude polynomial kernels. Building on OR-cross-compositions, we add a further tool (which we baptized "fractalism") in order to extend the range of problems to be addressed by OR-cross-compositions. The usefulness of our new technique is substantiated by resolving an open problem posed by Golovach and Thilikos [13], here specifically applying our technique to the NP-hard Length-Bounded Edge-Cut problem.

Next, we discuss in some more detail OR-(cross)-compositions. Roughly speaking, the idea behind an OR-composition for a parameterized problem is to encode the logical "or" of t instances with parameter value k into a single instance of the same problem with parameter value $k' = k^{O(1)}$. In particular, given t instances, the obtained instance is a yes-instance if and only if at least one of the given instances is a yes-instance. If an OR-composition is possible, then this excludes polynomial kernels. Whereas in OR-compositions one combines instances of an NP-hard parameterized problem into one instance of a parameterized problem, in OR-cross-compositions one combines instances of classical NP-hard problems into one instance of a parameterized problem (see Section 2 for details and formal definitions).

While for some problems, for example Longest Path with parameter solution size [5], a simple disjoint union yields the desired OR-composition, other problems seem to require involved constructions, for example Set Cover with parameter universe size [10]. Indeed, devising a cross-composition can be quite challenging and the task becomes even harder when considering several, seemingly orthogonal parameterizations at once. To illustrate the problem with such combined parameters, let us consider the problem Length-Bounded Edge-Cut (LBEC). Herein, an undirected graph $G = (V, E)$ with $s, t \in V$, and two integers $k, \ell \in \mathbb{N}$ are given, and the question is whether it is possible to delete at most k edges such that the shortest s-t path is of length at least ℓ. Using a simple branching algorithm, one can show that LBEC(k, ℓ) is fixed-parameter tractable for the combined parameter (k, ℓ) [13, 3]. To exclude the existence of a polynomial kernel for LBEC(k, ℓ), we would like to apply the OR-cross-composition framework to the problem, and as a natural candidate for the input problem we decide for LBEC itself.

A standard approach to applying the OR-cross-composition to a problem like LBEC would be to concatenate the input instances on the source and sink vertices, also referred to as "serial" composition. To this end, one needs some additional gadgets to ensure that only in one instance edges are deleted. This form of composition, however, induces a dependency of the second parameter ℓ on the number of instances, which is not allowed. Another standard approach is introducing a "global" sink and source vertex, and connecting all source vertices with the global source and all sink vertices, also referred to as a "parallel" composition. This form of composition would keep ℓ small enough, but induces a dependency of the first parameter k on the number of instances. Summarizing, the parameter k seems to ask for a serial composition and the parameter ℓ seems to ask for a parallel composition. For some problems using a tree as "instance selector" was helpful, see for example Bevern et al. [4] or Bazgan et al. [2]. The problem with trees is that they introduce small (constant-size) s-t cuts, which is problematic for Length-Bounded Edge-Cut. In this work, we introduce a fractal structure as instance selector which has the nice properties of trees but does not introduce small cuts. So, our fractal structure helps to exclude polynomial kernels for several problems.

Our contributions. Our main technical contribution is to introduce a family of graphs that we call T-fractals and that build on triangles. T-fractals feature a fractal-like structure, in the sense of self-similarity and scale-invariance. Using these T-fractals in OR-cross-compositions, we show that the following parameterized graph modification problems and several of their variants do not admit polynomial kernels (unless NP $\subseteq$ coNP / poly):

- Length-Bounded Edge-Cut(k, ℓ) (LBEC(k, ℓ)), where k is the number of edges to delete, and ℓ is the lower bound on the length of the shortest path.
- Minimum Diameter Edge Deletion(k, ℓ) (MDED(k, ℓ)), that is, given an undirected connected graph $G = (V, E)$ and two integers k, ℓ (the parameters), decide whether there are at most k edge deletions such that the remaining graph remains connected and has diameter at least ℓ.
- Directed Small Cycle Transversal(k, ℓ) (DSCT(k, ℓ)), that is, given a directed graph $G = (V, E)$ and two integers k, ℓ (the parameters), decide whether there are at most k edge deletions such that the remaining graph has no cycle of length smaller than ℓ.

Table 1 surveys our no-polynomial-kernel results and spots an open question.

Table 1 Survey of the concrete results of this paper (under the assumption that NP $\not\subseteq$ coNP / poly). PK stands for polynomial kernel and a "?" indicates that it is open whether a polynomial kernel exists. We remark that the no-polynomial-kernel results for LBEC(k, ℓ) on directed graphs still hold for directed acyclic graphs. Note that we claim without proof that, except for the planar variants, our proofs also transfer to the vertex deletion case, both for directed and undirected graphs.

Problem	edge deletion			
	directed		undirected	
	planar	general	planar	general
LBEC(k, ℓ)	No PK [Thm. 12]	No PK [Thm. 12]	No PK [Thm. 12]	No PK [Thm. 11]
MDED(k, ℓ)	No PK [Thm. 13]	No PK [Thm. 13]	No PK [Thm. 13]	No PK [Thm. 13]
DSCT(k, ℓ)	No PK [Thm. 14]	No PK [Thm. 14]	PK [25]	?

2 Preliminaries

Graph Theory. Let $G = (V, E)$ be a graph. For $C \subseteq V(G)$ ($C \subseteq E(G)$) we write $G - C$ for the graph G where all vertices (edges) in C are deleted. Let $s, t \in V(G)$. An edge set $C \subseteq E(G)$ is an *s-t edge cut* in G if the vertices s and t are disconnected in $G - C$. An s-t edge cut C is called *minimal* if $C \setminus \{e\}$ is not an s-t edge cut in G for all $e \in C$. An s-t edge cut C is called *minimum* if there is no s-t edge cut C' in G such that $|C'| < |C|$.

The length of a path (cycle) is the number of edges in the path (cycle). An s-t path is a path where the vertices s and t are the endpoints of the path. In directed graphs, an s-t path is a path where all arcs are directed toward t, and a cycle is a connected graph where every vertex has outdegree and indegree exactly one. The *diameter* of a graph G is the maximum length of any shortest v-w path over all $v, w \in V(G)$, $v \neq w$.

For $v, w \in V(G)$, we say we *merge* the vertices v and w if we add a new vertex vw to V as well as the edge set $\{\{vw, x\} \mid \{x, v\} \in E\} \cup \{\{vw, x\} \mid \{x, w\} \in E\}$ to E, and we delete the vertices v and w and all edges incident to v and w.

Parameterized Complexity. A *parameterized problem* is a set of instances $(\mathcal{I}, k)$ where $\mathcal{I} \in \Sigma^*$ for a finite alphabet Σ, and $k \in \mathbb{N}$ is the *parameter*. A parameterized problem L is *fixed-parameter tractable (fpt)* if it can be decided in $f(k) \cdot |\mathcal{I}|^{O(1)}$ time whether $(\mathcal{I}, k) \in L$, where f is a computable function only depending on k. We say that two instances $(\mathcal{I}, k)$ and $(\mathcal{I}', k')$ of parameterized problems P and P' are *equivalent* if $(\mathcal{I}, k)$ is YES for P if and only if $(\mathcal{I}', k')$ is YES for P'. A *kernelization* is an algorithm that, given an instance $(\mathcal{I}, k)$ of a parameterized problem P, computes in polynomial time an equivalent instance $(\mathcal{I}', k')$ of P (the *kernel*) such that $|\mathcal{I}'| + k' \leq f(k)$ for some computable function f only depending on k. We say that f measures the *size* of the kernel, and if $f \in k^{O(1)}$, we say that P admits a polynomial kernel. We remark that a decidable parameterized problem is fixed-parameter tractable if and only if it admits a kernel [8].

Given an NP-hard problem L, an equivalence relation $\mathcal{R}$ on the instance of L is a *polynomial equivalence relation* if (i) one can decide for any two instances in time polynomial in their sizes whether they belong to the same equivalence class, and (ii) for any finite set S of instances, $\mathcal{R}$ partitions the set into at most $(\max_{x \in S} |x|)^{O(1)}$ equivalence classes.

▶ **Definition 1.** Given an NP-hard problem L, a parameterized problem P, and a polynomial equivalence relation $\mathcal{R}$ on the instances of L, an *OR-cross-composition* of L into P (with respect to $\mathcal{R}$) is an algorithm that takes ℓ $\mathcal{R}$-equivalent instances $\mathcal{I}_1, \dots, \mathcal{I}_\ell$ of L and constructs

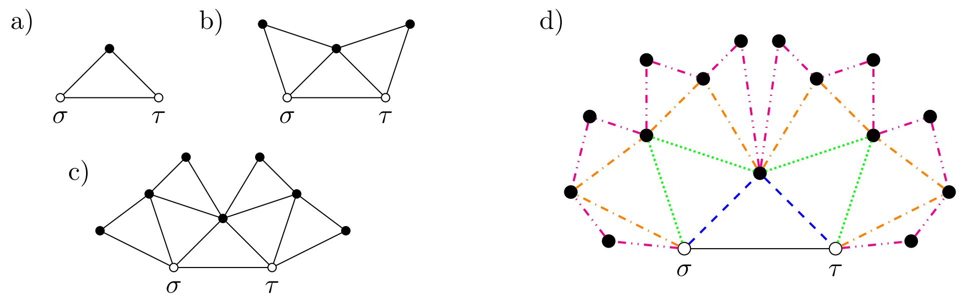

Figure 1 T-fractals a) $\triangle_1$, b) $\triangle_2$, c) $\triangle_3$, and d) $\triangle_4$. The two special vertices σ and τ are highlighted by empty circles. In $\triangle_4$ the different boundaries are highlighted by line-types (solid: boundary B_0; dashed: boundary B_1; dotted: boundary B_2; dash-dotted: boundary B_3; dash-dot-dotted: boundary B_4).

in time polynomial in $\sum_{i=1}^{\ell} |\mathcal{I}_\ell|$ an instance $(\mathcal{I}, k)$ such that

1. k is polynomially upper-bounded in $\max_{1 \le i \le \ell} |\mathcal{I}_i| + \log(\ell)$ and
2. $(\mathcal{I}, k)$ is YES for P if and only if there is at least one $\ell' \in [\ell]$ such that $\mathcal{I}_{\ell'}$ is YES for L.

If a parameterized problem P admits an OR-cross-composition for some NP-hard problem L, then P does not admit a polynomial kernel with respect to its parameterization, unless $\mathrm{NP} \subseteq \mathrm{coNP} / \mathrm{poly}$ [6]. We remark that we can assume that $\ell = 2^j$ for some $j \in \mathbb{N}$ since we can add trivial NO-instances from the same equivalence class to reach a power of two. We refer to the survey of Kratsch [17] for an overview on kernelization and lower bounds.

3 The "Fractalism" Technique

In this section, we describe our new technique based on *triangle fractals* (*T-fractals* for short). We provide a general construction scheme for cross-compositions using T-fractals. To this end, we first define T-fractals and then discuss several of their properties in Section 3.1. Subsequently, in Section 3.2 we present a "construction manual" for an application of T-fractals in cross-compositions.

Roughly speaking, a T-fractal can be constructed by iteratively putting triangles on top of each other, see Figure 1 for four examples.

▶ **Definition 2.** For $q \geq 1$, the *q-T-fractal* $\triangle_q$ is the graph constructed as follows:

(1) Set $\triangle_0 := \{\sigma, \tau\}$ with $\{\sigma, \tau\}$ being a "marked edge" with endpoints σ and τ, subsequently referred to as special vertices.
(2) Let F be the set of marked edges.
(3) For each edge $e \in F$, add a new vertex and connect it by two new edges with the endpoints of e, and mark the two added edges.
(4) Unmark all edges in F.
(5) Repeat (2)–(4) $q - 1$ times.

The fractal structure of $\triangle_q$ might be easier to see when considering the following equivalent recursive definition of $\triangle_q$: For the base case we define $\triangle_0 := \{\sigma, \tau\}$ as in Definition 2. Then, the q-T-fractal $\triangle_q$ is constructed as follows. Take two $(q-1)$-T-fractals $\triangle'_{q-1}$ and $\triangle''_{q-1}$, where σ', τ' and σ'', τ'' are the special vertices of $\triangle'_{q-1}$ and $\triangle''_{q-1}$, respectively. Then $\triangle_q$ is obtained by merging the vertices τ' and σ'' and adding the edge $\{\sigma', \tau''\}$. Set $\sigma = \sigma'$ and $\tau = \tau''$ as the special vertices of $\triangle_q$. We remark that we make use of the recursive structure in later proofs.

In the ith execution of (2)–(4) in Definition 2, we obtain 2^{i-1} many triangles. We say that these triangles have depth i. The *boundary* $B_i \subseteq E(\triangle_q)$, $i \in [q]$, are those edges of the triangles of depth i which are not edges of the triangles of depth $i-1$. As a convention, the edge $\{\sigma, \tau\}$ connecting the two special vertices σ and τ forms the boundary B_0. Refer to Figure 1 for an illustration of the boundaries in the T-fractal $\triangle_4$. Moreover, by construction, we obtain the following:

▶ **Observation 3.** *In every T-fractal, each boundary forms a σ-τ path, and all boundaries are pairwise edge-disjoint.*

Note that the boundary B_q contains $p = 2^q$ edges. Thus, the number of edges in $\triangle_q$ is $\sum_{i=0}^{q} 2^i = 2^{q+1} - 1 = 2 \cdot p - 1$. Further observe that all vertices of $\triangle_q$ are incident with the edges in B_q, and B_q forms a σ-τ path. Hence, $\triangle_q$ contains $p+1$ vertices.

Reducing the Weighted to the Unweighted Case. In the remainder of the paper, we focus on the unweighted case of T-fractals without multiple edges or loops. This is possible due to the following reduction of the weighted to the unweighted case. Equip the T-fractal with an *edge cost*, that is, the cost for deleting any edge in the T-fractal. If $c \in \mathbb{N}$ is the edge cost of $\triangle_q$, then we write $\triangle_q^c$ (we drop the superscript if $c = 1$). To reduce to the case with an unweighted, simple graph, we add $c-1$ further copies for each edge. Thus, to make two adjacent vertices non-adjacent, it requires c edge-deletions. To make the graph simple, we subdivide each edge. We remark that in this way we double the distances of the vertices in the original T-fractal. Thus, whenever we consider distances in the fractal with edge cost and the graph obtained by the reduction above, we have to take into account a factor of two.

3.1 Properties of T-Fractals

The goal of this subsection is to prove several properties of T-fractals that are used in later constructions. Some key properties of T-fractals appear in the context of σ-τ edge cuts in $\triangle_q$.

The minimum edge cuts in $\triangle_q$ will play a central role when using T-fractals in cross-compositions since the minimum edge cuts serve as instance selectors (see Section 3.2). First, we discuss the size and structure of the minimum edge cuts in $\triangle_q$.

▶ **Lemma 4.** *Every minimum σ-τ edge cut in $\triangle_q$ is of size $q+1$.*

Proof. Let C be a minimum σ-τ edge cut in $\triangle_q$. Note that the degrees of σ and τ are exactly $q+1$, and thus $|C| \leq q+1$. Moreover, the boundaries in $\triangle_q$ are pairwise edge-disjoint and each boundary forms a σ-τ path (Observation 3). Since $\triangle_q$ contains $q+1$ boundaries, it follows that there are at least $q+1$ disjoint σ-τ paths in $\triangle_q$. Menger's theorem [21] states that in a graph with distinct source and sink, the maximum number of disjoint source-sink paths equals the minimum size of any source-sink edge cut. Thus, by Menger's theorem, it follows that $|C| \geq q+1$. Hence $|C| = q+1$. ◀

From the fact that the boundaries are pairwise edge-disjoint and each boundary forms a σ-τ path, we can immediately derive the following from Lemma 4.

▶ **Corollary 5.** *Every minimum σ-τ edge cut in $\triangle_q$ contains exactly one edge of each boundary.*

In the following we describe a (hidden) dual structure in $\triangle_q$, that is, a complete binary tree with p leaves. We refer to Figure 2 for an example of the dual structure in $\triangle_3$. To talk

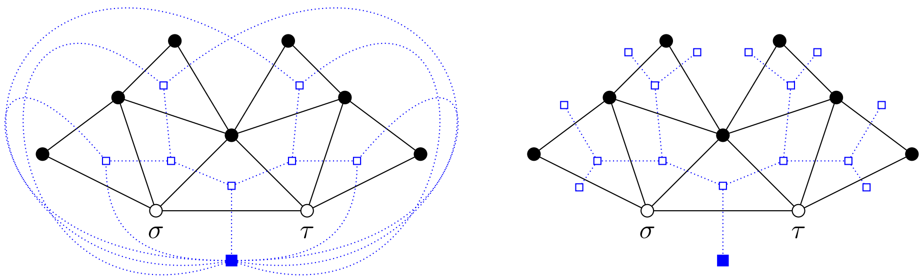

Figure 2 Left: The T-fractal $\triangle_3$ (circles and solid lines) and its dual graph (squares and dotted lines). The filled square is the vertex dual to the outer face in the dual graph. Right: The T-fractal $\triangle_3$ (circles and solid lines) and its dual structure T_3, illustrated by squares and dotted lines, where the filled square corresponds to the root of the dual structure.

about the dual structure by means of duality of plane graphs, we need a plane embedding of $\triangle_q$. Hence we assume that $\triangle_q$ is embedded as in Figure 1 (iteratively extended). By T_q we denote the dual structure in $\triangle_q$, where the vertex dual to the outer face is replaced by $p+1$ vertices (*split vertices*) such that each edge incident with the dual vertex is incident with exactly one split vertex. We consider the split vertex incident with the vertex dual to the triangle containing the edge $\{\sigma,\tau\}$ as the root vertex of the dual structure T_q. Thus, the other split vertices correspond to the leaves of the dual structure T_q. Note that the depth of a triangle one-to-one corresponds to the depth of the dual vertex in T_q.

Observe that there is a one-to-one correspondence between the edges in T_q and the edges in $\triangle_q$. The following lemma states duality of root-leaf paths in T_q and minimum σ-τ edge cuts in $\triangle_q$, demonstrating the utility of the dual structure T_q.

▶ **Lemma 6.** *There is a one-to-one correspondence between root-leaf paths in the dual structure T_q of $\triangle_q$ and minimum σ-τ edge cuts in $\triangle_q$. Moreover, there are exactly $p = 2^q$ pairwise different minimum σ-τ edge cuts in $\triangle_q$.*

Proof. Observe that each path from the root to a leaf in the dual structure T_q corresponds to a cycle in the dual graph. It is well-known that there is a one-to-one correspondence between minimal edge cuts in a plane graph and cycles in its dual graph [9, Proposition 4.6.1]. Herein, every cycle in the dual graph that "cuts" the edge $\{\sigma,\tau\}$ in $\triangle_q$ is a root-leaf path in T_q. Thus, the only minimal σ-τ edge cuts are those corresponding to the root-leaf paths. By the one-to-one correspondence of the depth of the triangles in $\triangle_q$ and the depth of the vertices in T_q, these edge cuts are of cardinality $q+1$. Hence, by Lemma 4, these edge cuts are minimum edge cuts.

Since $|B_q| = p$, there are exactly p leaves in T_q, and thus there are exactly p different root-leaf paths in T_q. It follows that the number of pairwise different minimum σ-τ edge cuts in $\triangle_q$ is exactly $p = 2^q$. ◀

Further, we obtain the following.

▶ **Lemma 7.** *Let C be a minimum σ-τ edge cut in $\triangle_q$. Let $\{x,y\} = C \cap B_q$, where x is in the same connected component as σ in $\triangle_q - C$. Then* $\operatorname{dist}(\sigma,x) + \operatorname{dist}(y,\tau) = q$ *in $\triangle_q - C$.*

Proof. We prove the lemma by induction on q. For the base case $q = 0$, observe that $C = \{\sigma,\tau\}$ and $\operatorname{dist}_{\triangle_0 - C}(\sigma,x) + \operatorname{dist}_{\triangle_0 - C}(y,\tau) = 0$.

For the induction step, assume that the statement of the lemma is true for $\triangle_{q-1}$. Now, let C be a minimum σ-τ edge cut in $\triangle_q$. Hence, $\{\sigma,\tau\} \in C$. Denote by u the (unique) vertex that is adjacent to the two special vertices σ and τ. Let $\triangle'_{q-1}$ and $\triangle''_{q-1}$ be the two $(q-1)$-T-subfractals of $\triangle_q$, so that $\triangle'_{q-1}$ ($\triangle''_{q-1}$) has the special vertices σ and u (u and τ). By Lemma 6, the minimum σ-τ edge cut C corresponds to a root-leaf path in T_q. Hence, $C' := C \setminus \{\sigma,\tau\}$ is either a subset of $E(\triangle'_{q-1})$ or of $E(\triangle''_{q-1})$. Assume w.l.o.g. that $C' \subseteq E(\triangle'_{q-1})$. It follows from the induction hypothesis that $\text{dist}_{\triangle'_{q-1}-C'}(\sigma,x)+\text{dist}_{\triangle'_{q-1}-C'}(y,u) = q-1$. Since $\text{dist}_{\triangle_q-C}(y,\tau) = \text{dist}_{\triangle'_{q-1}-C'}(y,u)+1$, it follows that $\text{dist}_{\triangle_q-C}(\sigma,x)+\text{dist}_{\triangle_q-C}(y,\tau) = q$. ◀

▶ Remark. By an inductive proof like the one of Lemma 7, one can easily show that the maximum degree Δ of $\triangle_q$ is exactly $2\cdot q$ for $q>0$. Moreover, due to Lemma 7, the diameter of $\triangle_q$ is bounded in $O(q)$.

Another observation on $\triangle_q$ is that any deletion of d edges increases the length of any shortest σ-τ path to at most $d+1$, unless the edge deletion forms a σ-τ edge cut.

▶ **Lemma 8.** *Let $D \subseteq E(\triangle_q)$ be a subset of edges of $\triangle_q$. If D is not a σ-τ edge cut, then there is a σ-τ path of length at most $|D|+1$ in $\triangle_q - D$.*

Proof. We prove the statement of the lemma by induction on q. For the induction base with $q=0$, observe that since D is not a σ-τ edge cut, it follows that $D=\emptyset$ and, hence, σ and τ have distance one.

For the induction step, assume that the statement of the lemma is true for $\triangle_{q-1}$. Now, let $D \subseteq E(\triangle_q)$ be a subset of edges of $\triangle_q$ such that D is not a σ-τ edge cut. If $\{\sigma,\tau\} \notin D$, then there is a σ-τ path of length one and the statement of the lemma holds. Now consider the case $\{\sigma,\tau\} \in D$. Denote by u the (unique) vertex that is adjacent to the two special vertices σ and τ. If $\{\sigma,\tau\} \in D$, then every σ-τ path in $\triangle_q - D$ contains u and hence $\text{dist}_{\triangle_q-D}(\sigma,\tau) = \text{dist}_{\triangle_q-D}(\sigma,u)+\text{dist}_{\triangle_q-D}(u,\tau)$. (If there is no σ-u-path or no u-τ-path in $\triangle_q - D$, then D is a σ-τ edge cut; a contradiction to the assumption of the lemma.) Now let $\triangle'_{q-1}$ and $\triangle''_{q-1}$ be the two $(q-1)$-T-subfractals of $\triangle_q$, so that $\triangle'_{q-1}$ ($\triangle''_{q-1}$) has the special vertices σ and u (u and τ). It follows that D can be partitioned into $D = D' \cup D'' \cup \{\sigma,\tau\}$ with $D' \subseteq E(\triangle'_{q-1})$ and $D'' \subseteq E(\triangle''_{q-1})$. By induction hypothesis, it follows that there is a σ-u path of length at most $|D'|+1$ in $\triangle'_{q-1} - D'$ and a u-τ path of length at most $|D''|+1$ in $\triangle''_{q-1} - D''$. Hence, there is a σ-τ path of length at most $|D'|+|D''|+2 = |D|+1$ in $\triangle_q - D$. ◀

We remark that there is a directed version of T-fractals. Herein, the edges of a T-fractal are directed in such a way that each boundary forms a directed σ-τ path. Note that the obtained graph is acyclic, and σ has no incoming arcs, and τ has no outgoing arcs. We further remark that all stated lemmas also hold for the directed T-fractal. The dual structure of a directed T-fractal is defined as the dual structure of the underlying undirected T-fractal. For more details we refer to the full version.

3.2 Application Manual for T-Fractals

The aim of this subsection is to provide a general guideline on how to use T-fractals in cross-compositions to obtain kernel lower bounds.

▶ **Construction 9.** *Given $p = 2^q$ instances $\mathcal{I}_1,\dots,\mathcal{I}_p$ of an NP-hard graph problem, where each instance $\mathcal{I}_i$ has a unique source vertex s_i and a unique sink vertex t_i.*

(i) *Equip $\triangle_q^c$ with some "appropriate" edge cost $c \in \mathbb{N}$.*

(ii) *Let $v_0,\dots,v_p$ be the vertices of the boundary B_q, labeled by their distances to σ in the σ-τ path corresponding to B_q (observe that $v_0=\sigma$ and $v_p=\tau$).*

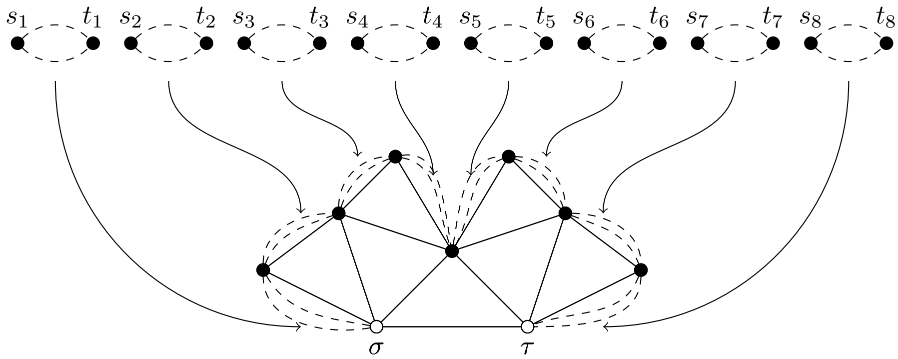

Figure 3 Illustration of Construction 9 with $p = 2^3 = 8$. The vertices $s_1, \ldots, s_8$ indicate the source vertices in the eight input instances, and $t_1, \ldots, t_8$ indicate the sink vertices in the eight input instances. We use dashed lines to sketch the input graphs.

(iii) *Incorporate each of the p graphs of the input instances into $\triangle_q^c$ as follows: for each $i \in [p]$, merge s_i with vertex v_{i-1} in $\triangle_q^c$ and merge t_i with v_i in $\triangle_q^c$.*

Refer to Figure 3 for an illustrative example of Construction 9.

In Construction 9, the T-fractal works as an instance selector by deleting edges corresponding to a minimum edge cut, which, by Lemma 4, is of size $q + 1$. Hence, each minimum edge cut costs $c \cdot (q + 1)$. The idea is that if we choose an appropriate value for c (larger than the budget in the instances $\mathcal{I}_1, \ldots, \mathcal{I}_p$) and an appropriate budget in the composed instance (e. g. $c \cdot (q + 1)$ plus the budget in the instances $\mathcal{I}_1, \ldots, \mathcal{I}_p$), then we can only afford to delete at most $q + 1$ edges in $\triangle_q^c$. Furthermore, if the at most $q + 1$ edges chosen to be deleted do not form a minimum σ-τ edge cut in $\triangle_q^c$, then, by Lemma 8, the shortest σ-τ path has length at most $q + 2$. Thus, by requiring in the composed instance that σ and τ have distance more than $q + 2$, we enforce that any solution for the composed instance contains a minimum σ-τ edge cut in $\triangle_q^c$. By Lemma 6, each such minimum edge cut corresponds to one root-leaf path in the dual structure T_q of $\triangle_q^c$. Observe that each leaf in the dual structure of $\triangle_q^c$ one-to-one corresponds to an attached source instance. Hence, with an appropriate choice of c, the budget in the composed instance, and the required distance between σ and τ, the T-fractal ensures that one instance is "selected". We say that a minimum σ-τ edge cut in $\triangle_q^c$ *selects* an instance $\mathcal{I}$ if the edge cut corresponds to the root-leaf path with the leaf corresponding to instance $\mathcal{I}$.

▶ **Observation 10.** *Every minimum edge cut C in $\triangle_q^c$ selects exactly one instance $\mathcal{I}$. Conversely, every instance $\mathcal{I}$ can be selected by exactly one minimum edge cut.*

We use Construction 9 in OR-cross-compositions to rule out the existence of polynomial kernels. We call this approach *fractalism*. In particular, we provide the source and the target problem, appropriate values for the edge cost c and the budget in the composed instance, and the required distance between the special vertices σ and τ.

We remark that there is a similar construction for directed graph problems using the directed T-fractal. Moreover, the construction with directed acyclic input graphs yields a directed acyclic graph. For more details, we refer to the full version.

4 Applications to Length-Bounded Cut Problems

In this section, we rule out the existence of polynomial kernels for several problems (and their variants) under the assumption that NP $\not\subseteq$ coNP / poly. To this end, we combine the framework of OR-cross-compositions with our fractalism technique as described in Section 3.2.

4.1 Length-Bounded Edge-Cut

Our first fractalism application is the LENGTH-BOUNDED EDGE-CUT problem [1], also known as the problem of finding bounded edge undirected cuts [13], or the SHORTEST PATH MOST VITAL EDGES problem [19, 3].

LENGTH-BOUNDED EDGE-CUT (LBEC)
Input: An undirected graph $G = (V, E)$, with $s, t \in V$, and two integers k, ℓ.
Question: Is there a subset $F \subseteq E$ of cardinality at most k such that $\operatorname{dist}_{G-F}(s,t) \geq \ell$?

The problem is NP-complete [16] and fixed-parameter tractable with respect to (k, ℓ) [13]. If k is at least the size of any s-t edge cut, then the problem becomes polynomial-time solvable by simply computing a minimum s-t edge cut. Thus, throughout this section, we assume that k is smaller than the size of any minimum s-t edge cut. The generalized problem where each edge is equipped with positive length remains NP-hard even on series-parallel and outerplanar graphs [1]. The directed variant with positive edge lengths remains NP-hard on planar graphs where the source and the sink vertex are incident to the same face [22]. Recently, Dvořák and Knop [11] showed that the problem can be solved in polynomial time on graphs of bounded treewidth. Here, we answer an open question [13] concerning the existence of a polynomial kernel with respect to the combined parameter (k, ℓ).

▶ **Theorem 11.** *Unless* NP $\subseteq$ coNP / poly*,* LENGTH-BOUNDED EDGE-CUT *parameterized by* (k, ℓ) *does not admit a polynomial kernel.*

Proof. We OR-cross-compose $p = 2^q$ instances of LBEC into one instance of LBEC(k', ℓ'). An instance $(G_i, s_i, t_i, k_i, \ell_i)$ of LBEC is called *bad* if $\max\{k_i, \ell_i\} > |E(G_i)|$ or $\min\{k_i, \ell_i\} < 0$. We define the polynomial equivalence relation $\mathcal{R}$ on the instances of LBEC as follows: two instances $(G_i, s_i, t_i, k_i, \ell_i)$ and $(G_j, s_j, t_j, k_j, \ell_j)$ of LBEC are $\mathcal{R}$-equivalent if and only if $k_j = k_i$ and $\ell_j = \ell_i$, or both are bad instances. Clearly, the relation $\mathcal{R}$ fulfills condition (i) of an equivalence relation (see Section 2). Observe that the number of equivalence classes of a finite set of instances is upper-bounded by the maximal size of a graph over the instances, hence condition (ii) holds. Thus, we consider p $\mathcal{R}$-equivalent instances $\mathcal{I}_i := (G_i, s_i, t_i, k, \ell)$, $i = 1, \ldots, p$. We remark that we can assume that $\ell \geq 3$, since otherwise LBEC is solvable in polynomial time by counting all edges connecting the source with the sink vertex. We OR-cross-compose into one instance $\mathcal{I} := (G, s, t, k', \ell')$ of LBEC(k', ℓ') with $k' = k^2 \cdot (\log(p) + 1) + k$ and $\ell' = \ell + \log(p)$ as follows.

Construction: Apply Construction 9 with edge cost $c = k^2$. In addition, set $s := \sigma$ and $t := \tau$. Let G denote the obtained graph.

Correctness: We show that $\mathcal{I}$ is a YES-instance if and only if there exists an $i \in [p]$ such that $\mathcal{I}_i$ is a YES-instance.

"⇐": Let $i \in [p]$ be such that $\mathcal{I}_i$ is YES. Following Observation 10 in Section 3.2, let C be the minimum s-t cut in $\triangle_q^c$ that selects instance $\mathcal{I}_i$. Recall that C is of size $q+1$ and that the edge cost equals k^2. Thus, the minimum s-t cut C has cost $(q+1) \cdot k^2 = (\log(p)+1) \cdot k^2$.

Note that after deleting the edges in C, the vertices s and t are only connected via paths through the incorporated graph G_i. Since $\mathcal{I}_i$ is YES, we can delete k edges (equal to the remaining budget) such that the distance of s_i and t_i in G_i is at least ℓ. Together with Lemma 7 in Section 3.1, such an additional edge deletion increases the length of any shortest s-t path in G to at least $\ell + \log(p) = \ell'$. Hence, $\mathcal{I}$ is a YES-instance.

"⇒": Suppose that one can delete at most k' edges in G such that each s-t path is of length at least ℓ'. Since the budget allows $\log(p)+1$ edge-deletions in $\triangle_q^c$, by Lemma 8 in Section 3.1, if we do not cut s and t in $\triangle_q^c$, then there is an s-t path of length $\log(p)+2$. Since $\ell \geq 3$, such an edge deletion does not yield a solution. Thus, in every solution of $\mathcal{I}$, a subset of the deleted edges forms a minimum s-t edge cut in $\triangle_q^c$ and thus, by Observation 10, selects an input instance.

Consider an arbitrary solution to $\mathcal{I}$, that is, an edge subset of $E(G)$ of cardinality at most k' whose deletion increases the shortest s-t path to at least ℓ'. Let $\mathcal{I}_i$, $i \in [p]$, be the selected instance. Note that any shortest s-t path contains edges in the selected instance $\mathcal{I}_i$. By Lemma 7, we know that the length of the shortest s-s_i path and the length of the shortest t_i-t path sum up to exactly $\log(p)$. It follows that the remaining budget of k edge deletions is spent in G_i in such a way that there is no path from s_i to t_i of length smaller than ℓ in G_i. Hence, $\mathcal{I}_i$ is a YES-instance. ◀

Using LBEC on planar graphs and on planar directed acyclic graphs as the input problem, fractalism yields the following (the proof is deferred to the full version).

▶ **Theorem 12.** *Unless* NP ⊆ coNP / poly*,* LENGTH-BOUNDED EDGE-CUT *on planar undirected graphs as well as on planar directed acyclic graphs parameterized by* (k, ℓ) *does not admit a polynomial kernel.*

4.2 Further Applications

We present two further problems (and their variants) to which the fractalism technique applies. First, we consider the following NP-hard [23] problem.

MINIMUM DIAMETER EDGE DELETION (MDED)
Input: A connected, undirected graph $G = (V, E)$, two integers k, ℓ.
Question: Is there a subset $F \subseteq E$ of cardinality at most k such that $G - F$ remains connected and $\operatorname{diam}(G - F) \geq \ell$?

Note that MDED on directed strongly connected graphs asks for a subset of arcs such that the graph after deleting the arcs remains strongly connected and has diameter at least ℓ. Our second NP-hard [15] problem is the following.

DIRECTED SMALL CYCLE TRANSVERSAL (DSCT)
Input: A directed graph $G = (V, E)$, two integers k, ℓ.
Question: Is there a subset $F \subseteq E$ of cardinality at most k such that there is no induced directed cycle of length at most ℓ in $G - F$?

Both problems are fixed-parameter tractable with respect to (k, ℓ) (see full version). The fractalism technique yields the following (the proofs are deferred to the full version).

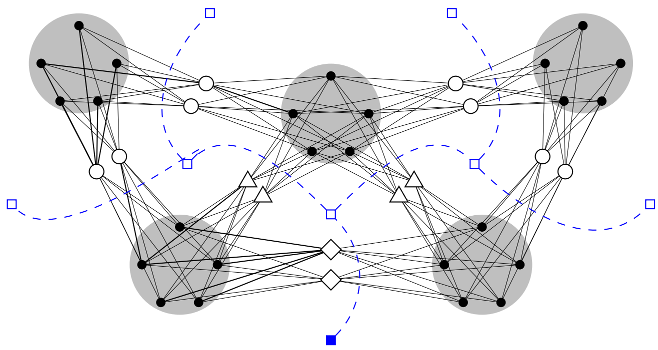

Figure 4 The vertex deletion variant $\triangle_2^{2;5}$ of T-fractals. Vertex types: empty diamonds belong to the boundary B_0, empty triangles belong to the boundary B_1, empty circles belong to the boundary B_2. The squares and dashed lines indicate the dual structure, where the filled square corresponds to the root. We highlighted vertices in gray-filled circles that correspond to the vertices in the edge-deletion variant $\triangle_2$.

▶ **Theorem 13.** *Unless* NP $\subseteq$ coNP / poly*,* Minimum Diameter Edge Deletion *on undirected, connected, planar graphs and on directed, strongly connected, planar graphs parameterized by* (k, ℓ) *does not admit a polynomial kernel.*

▶ **Theorem 14.** *Unless* NP $\subseteq$ coNP / poly*,* Directed Small Cycle Transversal *on planar directed graphs parameterized by* (k, ℓ) *does not admit a polynomial kernel.*

Like in the proof of Theorem 11, in the proofs of Theorems 13 and 14 we use LBEC as input problem and compose instances of LBEC using a T-fractal. The main difference is that we slightly modify the T-fractal. Roughly speaking, for MDED(k, ℓ) we append "long enough" paths to σ and τ, with endpoints σ' and τ' different to σ and τ. Those paths ensure that the only two vertices that can yield the required diameter are σ' and τ'. For DSCT(k, ℓ) we add the arc (τ, σ) to the directed T-fractal. Recall that the directed T-fractal is acyclic, and thus, every directed cycle in the composed graph contains the arc (τ, σ). For more details, we refer to the full version.

5 Conclusion

We start with briefly sketching how our technique can be adapted such that it also applies to the vertex deletion (instead of edge deletion) versions of the considered problems. Afterwards, we discuss future challenges and open problems.

Extension to Vertex-Deletion Variants. Most of our results can be transferred to the vertex deletion variants of the considered edge deletion problems as follows.

To this end, we modify the T-fractal as displayed in Figure 4:

First, subdivide each edge. Then, replace each vertex v in the original T-fractal by many pairwise non-adjacent vertices with the same neighborhood as v. The number of these introduced "false twins" is larger than the given budget such that the only way to disconnect vertices in the new fractal will be to delete vertices introduced from the subdivision of the edges. In this way, deleting a vertex in the new T-fractal corresponds to deleting an edge in the original fractal. This new fractal might not be planar anymore, but, as in the edge deletion variant, one can direct the edges in such a way that the obtained directed graph is acyclic.

We claim that the new T-fractal can be used in the same way as the original T-fractal in order to exclude polynomial kernels for vertex deletion variants of the problems discussed in this work – both in undirected and directed, but not for planar variants.

Outlook. We provided several case studies where our fractalism technique applies. It remains open to further explore the limitations and possibilities of our technique in more contexts. Table 1 in Section 1 presents an open question which should be clarified. Moreover, we could not settle the cases for vertex deletion problems when the underlying graphs are planar.

Acknowledgement. We thank Manuel Sorge (TU Berlin) for fruitful discussions.

References

1 Georg Baier, Thomas Erlebach, Alexander Hall, Ekkehard Köhler, Petr Kolman, Ondrej Pangrác, Heiko Schilling, and Martin Skutella. Length-bounded cuts and flows. *ACM Transactions on Algorithms*, 7(1):4, 2010. doi:10.1145/1868237.1868241.

2 Cristina Bazgan, Morgan Chopin, Marek Cygan, Michael R. Fellows, Fedor V. Fomin, and Erik Jan van Leeuwen. Parameterized complexity of firefighting. *Journal of Computer and System Sciences*, 80(7):1285–1297, 2014.

3 Cristina Bazgan, André Nichterlein, and Rolf Niedermeier. A refined complexity analysis of finding the most vital edges for undirected shortest paths. In *Proc. of the 9th International Conference on Algorithms and Complexity (CIAC 2015)*, volume 9079 of *LNCS*, pages 47–60. Springer, 2015. doi:10.1007/978-3-319-18173-8_3.

4 René van Bevern, Robert Bredereck, Morgan Chopin, Sepp Hartung, Falk Hüffner, André Nichterlein, and Ondřej Suchý. Parameterized complexity of dag partitioning. In *Proc. of the 8th International Conference on Algorithms and Complexity (CIAC'13)*, volume 7878 of *LNCS*, pages 49–60. Springer, 2013.

5 Hans L. Bodlaender, Rodney G. Downey, Michael R. Fellows, and Danny Hermelin. On problems without polynomial kernels. *Journal of Computer and System Sciences*, 75(8):423–434, 2009. doi:10.1016/j.jcss.2009.04.001.

6 Hans L. Bodlaender, Bart M. P. Jansen, and Stefan Kratsch. Kernelization lower bounds by cross-composition. *SIAM Journal on Discrete Mathematics*, 28(1):277–305, 2014. doi:10.1137/120880240.

7 Hans L. Bodlaender, Stéphan Thomassé, and Anders Yeo. Kernel bounds for disjoint cycles and disjoint paths. *Theoretical Computer Science*, 412(35):4570–4578, 2011.

8 Liming Cai, Jianer Chen, Rodney G. Downey, and Michael R. Fellows. Advice classes of parameterized tractability. *Annals of Pure and Applied Logic*, 84(1):119–138, 1997.

9 Reinhard Diestel. *Graph Theory*, volume 173 of *Graduate Texts in Mathematics*. Springer, 4th edition, 2010.

10 Michael Dom, Daniel Lokshtanov, and Saket Saurabh. Kernelization lower bounds through colors and ids. *ACM Transactions on Algorithms*, 11(2):13, 2014. doi:10.1145/2650261.

11 Pavel Dvořák and Dušan Knop. Parametrized complexity of length-bounded cuts and multi-cuts. In *Proc. of the 12th Annual Conference on Theory and Applications of Models of Computation (TAMC 2015)*, volume 9076 of *LNCS*, pages 441–452. Springer, 2015. doi:10.1007/978-3-319-17142-5_37.

12 Lance Fortnow and Rahul Santhanam. Infeasibility of instance compression and succinct PCPs for NP. *Journal of Computer and System Sciences*, 77(1):91–106, 2011. doi:10.1016/j.jcss.2010.06.007.

13 Petr A. Golovach and Dimitrios M. Thilikos. Paths of bounded length and their cuts: Parameterized complexity and algorithms. *Discrete Optimization*, 8(1):72–86, 2011. doi:10.1016/j.disopt.2010.09.009.

14 Jiong Guo and Rolf Niedermeier. Invitation to data reduction and problem kernelization. *ACM SIGACT News*, 38(1):31–45, 2007.

15 Venkatesan Guruswami and Euiwoong Lee. Inapproximability of feedback vertex set for bounded length cycles. *Electronic Colloquium on Computational Complexity (ECCC)*, 21:6, 2014.

16 Alon Itai, Yehoshua Perl, and Yossi Shiloach. The complexity of finding maximum disjoint paths with length constraints. *Networks*, 12(3):277–286, 1982. doi:10.1002/net.3230120306.

17 Stefan Kratsch. Recent developments in kernelization: A survey. *Bulletin of the EATCS*, 113:58–97, 2014.

18 Daniel Lokshtanov, Dániel Marx, and Saket Saurabh. Lower bounds based on the exponential time hypothesis. *Bulletin of the EATCS*, 105:41–72, 2011.

19 Kavindra Malik, Ashok K. Mittal, and Santosh K. Gupta. The k most vital arcs in the shortest path problem. *Operations Research Letters*, 8(4):223–227, 1989.

20 Dániel Marx. Parameterized complexity and approximation algorithms. *The Computer Journal*, 51(1):60–78, 2008.

21 Karl Menger. Über reguläre Baumkurven. *Mathematische Annalen*, 96(1):572–582, 1927.

22 Feng Pan and Aaron Schild. Interdiction problems on planar graphs. *Discrete Applied Mathematics*, 198:215–231, 2016.

23 Anneke A. Schoone, Hans L. Bodlaender, and Jan van Leeuwen. Diameter increase caused by edge deletion. *Journal of Graph Theory*, 11(3):409–427, 1987. doi:10.1002/jgt.3190110315.

24 David P. Williamson and David B. Shmoys. *The Design of Approximation Algorithms*. Cambridge University Press, 2011.

25 Ge Xia and Yong Zhang. On the small cycle transversal of planar graphs. *Theoretical Computer Science*, 412(29):3501–3509, 2011.