SIAM J. DISCRETE MATH.
Vol. 33, No. 3, pp. 1607–1636

# DISTANCE-PRESERVING GRAPH CONTRACTIONS[*]

AARON BERNSTEIN[†], KARL DÄUBEL[†], YANN DISSER[‡], MAX KLIMM[§], TORSTEN MÜTZE[†], AND FRIEDER SMOLNY[†]

**Abstract.** Compression and sparsification algorithms are frequently applied in a preprocessing step before analyzing or optimizing large networks/graphs. In this paper we propose and study a new framework contracting edges of a graph (merging vertices into supervertices) with the goal of preserving pairwise distances as accurately as possible. Formally, given an edge-weighted graph, the contraction should guarantee that for any two vertices at distance $d$, the corresponding supervertices remain at distance at least $\varphi(d)$ in the contracted graph, where $\varphi$ is a tolerance function bounding the permitted distance distortion. We present a comprehensive picture of the algorithmic complexity of the contraction problem for affine tolerance functions $\varphi(x) = x/\alpha - \beta$, where $\alpha \geq 1$ and $\beta \geq 0$ are arbitrary real-valued parameters. Specifically, we present polynomial-time algorithms for trees as well as hardness and inapproximability results for different graph classes, precisely separating easy and hard cases. Further we analyze the asymptotic behavior of contractions, and find efficient algorithms to compute (nonoptimal) contractions despite our hardness results.

**Key words.** spanner, contraction, distance oracle, graph compression

**AMS subject classifications.** 68R10, 05C85

**DOI.** 10.1137/18M1169382

## 1. Introduction.

When dealing with large networks, it is often beneficial to compress or sparsify the data to manageable size before analyzing or optimizing the network directly. To be useful, a meaningful compression should represent salient features of the original network with good approximation, while being much smaller in size. In this paper, we focus on a compression of undirected edge-weighted graphs that approximately maintains all distances between vertices in the graph.

In this context, an extensively studied concept is *spanners* (e.g., [1, 3, 7, 33]). Given an undirected graph $G = (V, E)$ and real numbers $\alpha \geq 1$ and $\beta \geq 0$, a subgraph $H = (V, E')$, $E' \subseteq E$, is an $(\alpha, \beta)$-*spanner of* $G$ if $\mathrm{dist}_H(u, v) \leq \alpha \cdot \mathrm{dist}_G(u, v) + \beta$ holds for all $u, v \in V$. While the number of edges in a spanner may be much smaller than that of the original graph, the number of vertices is the same for both, leaving further potential for compression untapped. For illustration, consider the road network of Europe with about 50 million vertices [5], any spanner of which must again have about 50 million vertices and edges. However, to approximately represent distances in Europe's road network one may also merge nearby vertices into supervertices, thus achieving a much better compression of the network. This is akin to the visual process

[*] Received by the editors February 6, 2018; accepted for publication (in revised form) May 30, 2019; published electronically September 5, 2019. An extended abstract of this work appeared in the Proceedings of the 9th Innovations in Theoretical Computer Science Conference (ITCS), 2018, Schloss Dagstuhl–Leibniz-Zentrum für Informatik, Wadern, Germany, 2018, 51, [10].
https://doi.org/10.1137/18M1169382

**Funding:** The second, fourth, fifth, and sixth authors were funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy–The Berlin Mathematics Research Center MATH+ (EXC-2046/1, project ID: 390685689). The third author was supported by the Excellence Initiative of the German Federal and State Governments and the Graduate School CE at TU Darmstadt.

[†] Institut für Mathematik, TU Berlin, Berlin (bernstei@gmail.com, daeubel@math.tu-berlin.de, muetze@math.tu-berlin.de, smolny@math.tu-berlin.de).

[‡] Department of Mathematics, Graduate School CE, TU Darmstadt, Darmstadt, Germany (disser@mathematik.tu-darmstadt.de).

[§] Wirtschaftswissenschaftliche Fakultät, HU Berlin, Berlin (max.klimm@hu-berlin.de).

of zooming out of a graphical representation of the map, where neighbored vertices fade into each other and edges between merged vertices vanish. At a large enough zoom level, the entire network merges into a single vertex.

In this paper we propose and study a new framework for contracting networks that formalizes this intuitive idea and makes it applicable to general graphs. Specifically, we study a contraction problem on graphs where a subset of edges $C \subseteq E$ is contracted. We denote by $G/C$ the resulting simple graph obtained from $G$ by contracting the edges in $C$ and by deleting resulting loops and multiple edges, keeping only the minimum length edge between any two vertices. For any two vertices in $G$, we compare their distance in $G$ with the distance of the corresponding supervertices in $G/C$.

It is interesting to contrast this concept with graph spanners. When constructing a spanner, the length of the removed edges is implicitly set to $\infty$, resulting in an overall increase of distances. On the other hand, a contraction implicitly sets the length of the contracted edges to zero, leading to an overall decrease of distances. For both problems, the ultimate goal is to reduce the complexity of the network while maintaining an approximation guarantee on the distances.

The following example shows that contractions may be better suited than spanners to achieve this goal. In a subgraph with small radius, a spanner can at best result in a spanning tree of the same order, while a contraction can reduce the whole subgraph to a single vertex, while entailing a multiplicative distance distortion of similar magnitude. In addition, the contraction may also merge many edges entering the contracted subgraph. Clearly, the objective here is to maximize the total number of contracted and deleted edges, as this minimizes the memory required to represent the resulting network in a computer (using, e.g., adjacency lists).

Given the results presented in this paper and the known results for spanners (discussed in detail below), we further believe that the combination of spanners and contractions is very powerful, promising, and flexible. As the former only increases and the latter only decreases the distances, the respective distortion guarantees provably also hold for the overall distortion. In fact, both effects may even compensate each other. This is true *regardless* of the order in which both compression operations are applied, even when they are applied repeatedly.

In order to measure the distance distortion of the contraction, we assume a nondecreasing tolerance function $\varphi\colon \mathbb{R} \to \mathbb{R}$, similar to the corresponding function for spanners; see, e.g., [7]. We are interested in computing contractions that preserve distances in the following sense: For any two vertices $u$ and $v$ at distance $d$ in $G$, the distance of the corresponding vertices in the contracted graph $G/C$ must be at least $\varphi(d)$. If this condition is satisfied, we call $C$ a *$\varphi$-distance preserving contraction*, or *$\varphi$-contraction* for short. Formally, the algorithmic problem CONTRACTION considered in this paper is to compute for a given graph $G = (V, E)$ with edge lengths $\ell\colon E \to \mathbb{R}_{>0}$ and a given tolerance function $\varphi$, a $\varphi$-contraction $C \subseteq E$ such that the number of contracted and deleted edges is maximized. We are specifically interested in the case where the tolerance function $\varphi$ is an affine function $\varphi(x) = x/\alpha - \beta$ for real-valued parameters $\alpha \geq 1$ and $\beta \geq 0$. We then simply write *$(\alpha, \beta)$-contraction* instead of $\varphi$-contraction. See Figure 1 for some example instances of the problem CONTRACTION.

When considering the case of a purely multiplicative error ($\beta = 0$), a slight subtlety has to be taken into account. Specifically, for a graph with positive edge lengths it is not feasible to contract a single edge. Therefore, we propose a slight modification of our original model: We say that a set $C \subseteq E$ of edges of $G$ is a *weak $\varphi$-distance preserving contraction*, or *weak $\varphi$-contraction* for short, if it does not contract the entire graph and, for any two vertices $u$ and $v$ at distance $d$ in $G$, the

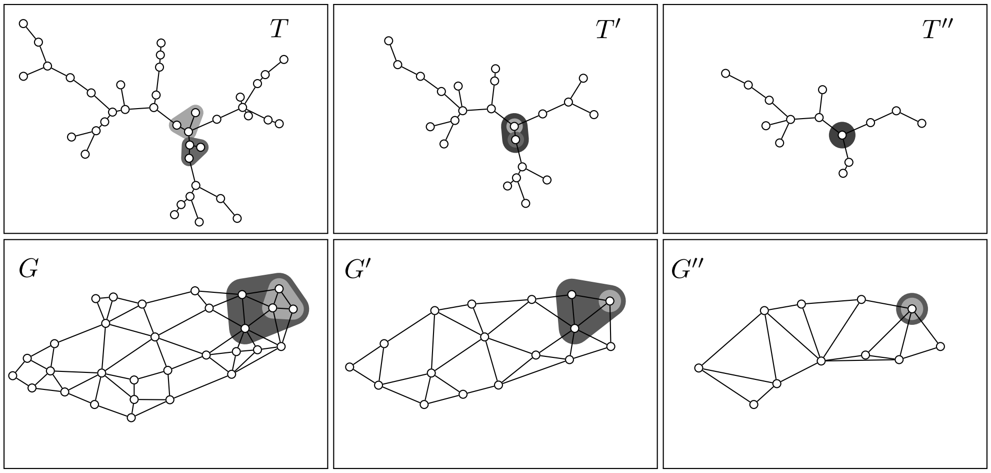

Fig. 1. *Top: two iterations of* Contraction *with* $\varphi(x) = 4x/5 - 3$ *on a tree; bottom: two iterations of* Contraction *with* $\varphi(x) = 3x/4 - 3$ *on a planar graph. Distances are geometric and some contracted sets of vertices are highlighted.*

distance of the corresponding vertices in $G/C$ is either zero or at least $\varphi(d)$. We will refer to the corresponding algorithmic problem as Weak Contraction. Put differently, in a weak contraction, the distances between different supervertices satisfy the given distortion guarantee, but for vertices belonging to the same supervertex, no guarantee is given.

**1.1. Our results.** In this paper, we present a comprehensive picture of the algorithmic complexity of the described contraction problems. Recall that we are given an input graph with edge lengths and tolerance function $\varphi$, and our goal is to compute a (weak) contraction that maximizes the total number of contracted and deleted edges. Our main results concern affine tolerance functions $\varphi(x) = x/\alpha - \beta$ with parameters $\alpha \geq 1$ and $\beta \geq 0$. For the reader's convenience, our results are summarized in Tables 1 and 2. Within the tables and throughout this paper, $n$ and $m$ denote the number of vertices and edges, respectively, of the input graph under consideration.

**Algorithmic results.** We first present a linear time greedy algorithm for Contraction on cycles with unit lengths for general $\alpha$ and $\beta$ (Theorem 2). The algorithm is inspired by linear programming rounding techniques.

We also develop dynamic programming algorithms solving Contraction and Weak Contraction on trees in time $\mathcal{O}(n^3)$ or $\mathcal{O}(n^5)$, respectively (Theorems 3 and 4). These dynamic programs compute optimal solutions on subtrees, in the latter case combining several Pareto optimal solutions in a two-dimensional parameter space (hence the larger running time).

Note that instead of maximizing the number of contracted and deleted edges, we could optimize for $\alpha$ or $\beta$ while fixing the other parameters. The resulting problems are polynomially equivalent to our setting, via binary search over one of the parameters.

**Hardness results.** We complement these algorithms by several hardness results. First we consider the purely additive case where $\alpha = 1$. We show that here both Contraction and Weak Contraction are NP-hard on cycles for any fixed $\beta > 0$, by a reduction of a variant of Partition (Theorem 5). As mentioned before, both

Table 1
*Overview of algorithmic and hardness results presented in this paper.*

| Problem | Graph classes | | |
|---|---|---|---|
| | Tree | Cycle | General |
| Contraction | | | |
| addit. ($\alpha$=1), unit lg. | | | $m^{\frac{1}{2}-\varepsilon}$-inapx.[a] [Thm. 8] |
| affine ($\alpha, \beta$), unit lg. | | $\mathcal{O}(n)$ [Thm. 2] | |
| addit. ($\alpha$=1) | | NP-hard [Thm. 5] | $n^{1-\varepsilon}$-inapx. [Thm. 7] |
| affine ($\alpha, \beta$) | $\mathcal{O}(n^3)$ [Thm. 3] | | |
| Weak Contraction | | | |
| additive ($\alpha$=1) | | NP-hard[b] [Thm. 5] | |
| affine ($\alpha, \beta$) | $\mathcal{O}(n^5)$ [Thm. 4] | | $n^{1-\varepsilon}$-inapx.[c] [Thm. 10] |

[a] even for bipartite graphs and $\beta = 1$.

[b] also NP-hard for planar graphs with arb. large girth, $(\alpha, \beta) = (2, 0)$, and unit lg. ($\ell = 1$) [Thm. 9].

[c] even if $(\alpha, \beta) = (3/2, 0)$.

Table 2
*Overview of asympotic bounds presented in this paper.*

| Contraction with unit lg. ($\ell$=1) | # of edges in $G/C$ | Time | Reference |
|---|---|---|---|
| $(\alpha, \beta) = (2k-1, 1)$ | $n^{1+1/k}$ | $\mathcal{O}(m)$ | [Thm. 11] |
| $(\alpha, \beta) = (2\log_2 n - 1, 1)$ | $2n$ | $\mathcal{O}(m)$ | [Cor. 12] |
| $(\alpha, \beta) = (k-1, 1)$ | $\Omega(n^{1+1/k})$ | — | [Thm. 13] |
| $(\alpha, \beta) = (1, k)$ | $m - km/(2n)$ | $\mathcal{O}(m)$ | [Thm. 14(i)] |
| $(\alpha, \beta) = (1, k)$ | $\mathcal{O}(n^2/k)$ | $\mathcal{O}(m)$ | [Thm. 14(ii)] |
| $(\alpha, \beta) = (1, \mathcal{O}(1))$ | $\Omega(n^{4/3-o(1)})$ | — | [1] |
| Contraction with unit lg. ($\ell$=1) and min. degree $D$ | # of vertices in $G/C$ | Time | Reference |
| $(\alpha, \beta) = (5, 1)$ | $n/D$ | $\mathcal{O}(m)$ | [Thm. 15] |
| $(\alpha, \beta) = (k, 1)$ | $n/((k+1)D)$ | — | [Thm. 16] |

problems can be solved efficiently on graphs without cycles, and there is a linear time algorithm for Contraction on cycles with unit lengths. By reductions from Clique we show that both the general as well as the unit lengths case of Contraction with $\alpha = 1$ are hard to approximate within factors of $n^{1-\varepsilon}$ or $m^{1/2-\varepsilon}$, respectively (Theorems 7 and 8).

Further we consider the purely multiplicative case where $\beta = 0$ (here Contraction is trivial). We show that in this case Weak Contraction is NP-hard on planar graphs with arbitrarily large girth and unit length edges by a reduction from a special case of Planar 3SAT (Theorem 9). Since these graphs are locally treelike, this result constitutes another rather sharp separation from the polynomially solvable tree case. Furthermore, we show that the problem is hard to approximate within a factor of $n^{1-\varepsilon}$ by a reduction from Independent Set (Theorem 10).

**Asymptotic bounds.** We now discuss our asymptotic bounds for contractions. In this setting, we are interested in (nonoptimal) contractions for graphs with unit lengths that can be computed efficiently despite the above-mentioned hardness results. We prove that for any $k \geq 1$, any graph $G$ has a $(2k-1, 1)$-contraction $C$ such that $G/C$ has at most $n^{1+1/k}$ edges, and such a contraction can be computed in time $\mathcal{O}(m)$ (Theorem 11) by successively growing clusters around center vertices. Assuming Erdős' girth conjecture, we show a corresponding (not tight) lower bound (Theorem 13).

For a purely additive error, we observe two simple $(1, k)$-contractions that can be computed in $\mathcal{O}(m)$ time (Theorem 14). We show that for any even integer $0 \leq k \leq n$, the edges incident to the $k/2$ vertices of highest degrees form a $(1, k)$-contraction with objective value at least $km/(2n)$, which is asymptotically best possible for paths. Another $(1, k)$-contraction $C$ is implicitly used by Bernstein and Chechik in their faster deterministic algorithm for dynamic shortest paths in dense graphs [9]. For any number $0 < k \leq n$, it consists of the edges incident to two vertices of degree at least $n/k$, and $G/C$ has $\mathcal{O}(n^2/k)$ edges. Both of these contractions can be computed in $\mathcal{O}(m)$ time. Further we note that the main result in [1] implies that for all $\varepsilon > 0$, any contraction $C$ such that $G/C$ has $\mathcal{O}(n^{4/3-\varepsilon})$ edges does not admit a constant additive error.

One possible advantage of contraction compared to spanners is the potentially significant reduction of *vertices* as well as edges, e.g., reducing the complexity of performing algorithmic tasks in the smaller graph. To ground this intuition, we exhibit a contraction that significantly reduces the number of vertices in any graph with minimum degree $D$ to $\mathcal{O}(n/D)$ (Theorem 15). We also present a lower bound (Theorem 16) showing that we cannot guarantee $o(n/D)$ vertices, even if we allow larger approximation error.

**1.2. Comparison with previous results.** There are several models aiming to compress graphs while preserving distances. They differ by their choice of compression operation, such as replacing the graph by a subgraph or minor, and by whether the aim is to preserve all or only certain distances.

As discussed before, graph spanners are a concept closely related to contractions, where the length of removed edges is set to $\infty$ rather than to 0. Our results highlight further intrinsic similarities of the two models. Like contractions, spanners are NP-hard to compute optimally (see [31, 33]). While the spanner literature considers the problem of minimizing the number of remaining edges, we analyze the objective of maximizing the number of contracted edges, prohibiting a direct comparison of the respective inapproximability results. We note however that approximation algorithms for spanner problems have been studied extensively, even though strong lower bounds are known. For instance, computing $(2, 0)$-spanners in unweighted graphs is $\Theta(\log n)$-hard to approximate (see [27, 28]); for further references, see, e.g., [17].

Despite these negative results, it is still possible to obtain powerful asymptotic guarantees in both models. In particular, our $(2k-1, 1)$-contraction with $\mathcal{O}(n^{1+1/k})$ edges for unweighted graphs has a clear analogy to the classic $(2k-1, 0)$-spanner with the same number of edges [3] (note that the additive error of 1 in our result is strictly necessary, as discussed above). There is, however, a major difference between the two results: Whereas the $(2k-1, 0)$-spanner can trivially be shown to be optimal assuming Erdős' girth conjecture, applying this conjecture to the contraction model only yields a lower bound of $n^{1+1/(2k)}$ edges for a $(2k-1, 1)$-contraction. Closing this gap thus remains as an interesting open problem in the contraction model, whose solution would likely yield further insight into the relationship to spanners.

Halperin and Zwick showed how an optimal $(2k-1, 0)$-spanner can be constructed in linear time (see [8]). We achieve the same running time for our $(2k-1, 1)$-

contraction. It is interesting to note that the clustering yielding our $(2k-1,1)$-contraction was previously used in [33] to obtain a $(4k+1,0)$-spanner of the same asymptotic density.

There are also spanner results that significantly sparsify unweighted graphs at the cost of a purely additive error, as a (1,2)-spanner with $\mathcal{O}(n^{3/2})$ edges [2], or a (1,6)-spanner with $\mathcal{O}(n^{4/3})$ edges [7]. We do not know if analogous results are possible in the contraction model. The incompressibility result in [1] mentioned above implies the same lower bound for spanners as for contractions and every other distance oracle with additive error: For every $\varepsilon > 0$ any spanner of size $\mathcal{O}(n^{4/3-\varepsilon})$ does not admit a constant additive error. Finally, for spanners there are results that combine multiplicative and additive error, such as the $(k, k-1)$-spanner of [7].

Gupta [24] considered the problem of approximating a tree metric on a subset of the vertices by another tree, and gave a linear time algorithm computing an 8-approximation. As Chan et al. [15] observed later, on complete binary trees a solution of minimum distortion is always achieved by a minor (with possibly different edge lengths) of the input tree, so this seems to be the first investigation of contractions that approximate graph distances. Krauthgamer, Nguyen, and Zondiner [29] considered an extension to general graphs, studying the size of minors preserving all distances between a given terminal set of fixed size. Cheung, Goranci, and Henzinger [16] introduced a multiplicative distortion to this model. As here no two terminals may be merged, these approaces cannot compress a graph at all if every vertex is a terminal.

The *pairwise preservers* due to Coppersmith and Elkin [19] combine spanners with the aim of preserving only terminal distances. Given a graph $G$ and a set of $k$ terminal pairs, a pairwise preserver is a spanning subgraph inducing exactly the same terminal distances as $G$. Coppersmith and Elkin [19] proved that for every undirected weighted graph there exists a pairwise preserver of size $\mathcal{O}(n+n^{1/2}k)$. Furthermore, they showed that every directed weighted graph has a pairwise preserver of size $\mathcal{O}(nk^{1/2})$. For the special case of undirected unweighted graphs, Bodwin and Vassilevska Williams [14] showed the existence of a pairwise preserver with $\mathcal{O}(n^{2/3}k^{2/3}+nk^{1/3})$ edges. Recently, Bodwin [13] proved that any directed weighted graph has a pairwise preserver of size $\mathcal{O}(n+n^{2/3}k)$.

**1.3. Further related work.** The preservation of graph properties other than distances has been studied as well. Biedl, Brejová, and Vinař [12] considered contractions in capacitated networks with the goal of maintaining the maximum flow in the network. Here an edge $e$ is called *useless*, if for every capacity function there is a maximum flow not using $e$. Biedl, Brejová, and Vinař showed that finding all useless edges is NP-complete, but solvable in $\mathcal{O}(n^2)$ time on certain planar graphs. For undirected networks, Misiołek and Chen [32] gave an algorithm finding all useless edges in $\mathcal{O}(n+m)$ time. Zhou, Mahler, and Toivonen [35] considered a more general model aiming to maintain the quality of paths with respect to any given function, e.g., distance or capacity. They investigated strategies of removing edges, without decreasing the quality of the best path between any pair of vertices.

Graph simplification problems have also been studied in several other contexts, and we conclude this section by mentioning two such examples:

Hübler et al. [25] studied a problem related to graph mining, examining how to choose an induced subgraph with a given number of vertices and with similar topological properties as the input graph. Numerous papers investigate, directly or as a tool, sparsifiers that preserve the effective resistance between certain or all pairs of vertices; see, e.g., [18, 20, 21, 22, 30].

**1.4. Outline of this paper.** In section 2 we introduce important definitions and notations that will be used throughout this paper. In section 3 we present the greedy algorithm for solving CONTRACTION with unit lengths on cycles. In section 4 we discuss efficient dynamic programming algorithms for CONTRACTION and WEAK CONTRACTION on trees. Sections 5 and 6 are devoted to our hardness results, focussing on the cases of purely additive and multiplicative error, respectively. In section 7 we present our asymptotic results on contractions. For the sake of brevity, in this paper we omit all proofs on the WEAK CONTRACTION problem. They can be found in the preprint version available on arXiv [11].

## 2. Preliminaries.

Throughout this paper we consider simple undirected graphs $G$ (without parallel edges or loops). We let $V(G)$ and $E(G)$ denote the vertex and edge set of $G$, respectively, and we define $n(G) := |V(G)|$ and $m(G) := |E(G)|$. If the context is clear, we simply write $V$, $E$, $n$, and $m$. We also use the notation $[n] := \{1, 2, \dots, n\}$. We assume that $G$ is connected, otherwise the contraction problem can be solved independently for each connected component. Edge lengths are given by a function $\ell \colon E \to \mathbb{R}_{>0}$. The *distance* $\mathrm{dist}_\ell(u, v)$ between two vertices $u$ and $v$ is the length of a shortest path between $u$ and $v$ in $G$ with respect to $\ell$.

Given a subset of edges $C \subseteq E$, we denote the resulting simple graph obtained from $G$ by contracting the edges in $C$, deleting resulting loops and keeping only the minimum length edge between any two vertices by $G/C$. We denote the number of deleted loops and multiedges by $\Delta(C)$ (thus $m(G/C) = m(G) - |C| - \Delta(C)$). Instead of contracting a set $C \subseteq E$ of edges in $G$, setting their edge lengths to zero has the same effect on the distances in the resulting graph. This is somewhat cleaner conceptually, so we will often adopt this viewpoint. Specifically, we let $\ell_C$ be the new length function that assigns 0 to every edge in $C$, and that is equal to the original edge lengths $\ell$ on the edges $E \setminus C$.

A *tolerance function* is a nondecreasing function $\varphi \colon \mathbb{R} \to \mathbb{R}$. Roughly speaking, this function describes by how much the distance between two vertices may drop when contracting edges (i.e., setting edge lengths to zero). Formally, given a graph $G$ with edge lengths $\ell$ and a tolerance function $\varphi$, we say that a subset of edges $C \subseteq E$ is a $\varphi$-contraction, if

$$\mathrm{dist}_{\ell_C}(u, v) \geq \varphi(\mathrm{dist}_\ell(u, v)) \tag{1}$$

holds for any two vertices $u$ and $v$ in $G$. Similarly, we say that $C$ is a weak $\varphi$-contraction, if any two vertices $u$ and $v$ satisfy relation (1) or the relation $\mathrm{dist}_{\ell_C}(u, v) = 0$, and if the graph $(V, C)$ is disconnected (equivalently, if $G/C$ is not a single vertex). The last condition prevents solutions $C \subseteq E$ for which the graph is contracted to a single vertex. If $\varphi(x) = x/\alpha - \beta$, then we simply write (weak) $(\alpha, \beta)$-contraction instead of (weak) $\varphi$-contraction.

An *instance* of the problem CONTRACTION or WEAK CONTRACTION is a triple $(G, \ell, \varphi)$, where $G$ is the underlying graph, $\ell$ the length function, and $\varphi$ the tolerance function, and the objective is to find a (weak) $\varphi$-contraction $C \subseteq E$, such that

$$\Phi(C) := |C| + \Delta(C) = m(G) - m(G/C) \tag{2}$$

is maximized. This quantity equals the number of edges we save when going from $G$ to $G/C$. Note that on trees we have $\Phi(C) = |C|$ for any (weak) contraction $C$, whereas on general graphs we have $\Phi(C) \geq |C|$.

(Weak) Contraction
**Input:** A graph $G = (V, E)$ with edge lengths $\ell\colon E \to \mathbb{R}_{>0}$ and a nondecreasing function $\varphi\colon \mathbb{R} \to \mathbb{R}$.
**Output:** A ) $\varphi$-contraction $C \subseteq E$ maximizing $\Phi(C)$.

In this context we sometimes refer to a set of edges that forms a (weak) contraction as a *feasible* solution, and to a (weak) contraction of maximum value $\Phi(C)$ as an *optimal* solution.

We begin by proving that our contraction model behaves nicely when contracting edges in phases, i.e., the total error is simply the error accumulated over the contraction phases (but not more). To state this result we denote the composition of tolerance functions $\varphi$ and $\psi$ as $(\psi \circ \varphi)(x) := \psi(\varphi(x))$.

Theorem 1. *Let $C$ be a (weak) $\varphi$-contraction for $G$, and let $C'$ be a (weak) $\psi$-contraction for $G/C$. Then $C \cup C'$ is a (weak) $(\psi \circ \varphi)$-contraction for $G$.*

*Proof.* We only prove the statement for contractions $\varphi$ and $\psi$. The proof for weak contractions works analogously. Let $\ell$ denote the edge lengths of $G$ and consider a pair of vertices $u, v \in V(G)$. Then we have $\operatorname{dist}_{\ell_{C\cup C'}}(u, v) \geq \psi(\operatorname{dist}_{\ell_C}(u, v))$ by the definition of $C'$ and $\operatorname{dist}_{\ell_C}(u, v) \geq \varphi(\operatorname{dist}_\ell(u, v))$ by the definition of $C$. Combining these inequalities and using that $\psi$ is nondecreasing we obtain $\operatorname{dist}_{\ell_{C\cup C'}}(u, v) \geq \psi(\varphi(\operatorname{dist}_\ell(u, v)))$, as desired. □

Note that Theorem 1 only concerns the *feasibility* of repeated contractions, but is not about their *optimality* when searching for contractions of maximum cardinality. With respect to solution quality, contracting in phases may be arbitrarily bad: Consider a star with $k$ unit length edges and additive tolerance functions $\varphi(x) = \psi(x) = x - 1$. An optimum $(\psi \circ \varphi)$-contraction contains all $k$ edges, whereas finding an optimal $\varphi$-contraction $C$ and then an optimal $\psi$-contraction of $G/C$ allows contracting only one edge in each phase, leading to a $(\psi \circ \varphi)$-contraction of value 2.

## 3. A greedy algorithm for cycles with unit length edges.

In this section we consider the special case of contracting a cycle $C_n$ with $n$ vertices and unit length edges $\ell = 1$ and the tolerance function $\varphi(x) = x/\alpha - \beta$, $\alpha \geq 1$, $\beta \geq 0$. For this case we present a greedy algorithm running in linear time. The main purpose of this result is to clearly separate the polynomially solvable cases of Contraction from the NP-hard cases, and the case of a cycle with unit length edges precisely forms this boundary on the polynomially solvable side. Recall in this context that we can solve Contraction in polynomial time on any tree (this will be proved in section 4.1 below), and that Contraction is NP-hard already on a cycle for $\alpha = 1$ (with arbitrary edge lengths; we will show this in section 5.1 below).

We first argue that on a cycle it is equivalent to maximizing the number of contracted edges $|C|$ or to maximizing our objective function $\Phi(C)$ defined in (2). This is because the set of pairs $(|C|, \Phi(C))$ for all feasible contractions $C$ in a cycle $G = C_n$ is given by $\{(1, 1), (2, 2), \ldots, (n-3, n-3), (n-2, n-1), (n-1, n), (n, n)\}$, so it forms a monotone function, implying that maximizing either one of the two quantities is equivalent. Based on this argument, for the rest of this section we consider maximizing the number $|C|$ of contracted edges.

Observe that a solution $C \subseteq E(C_n)$ ($C_n$ is the cycle we want to contract and $C$ is the set of edges to be contracted) for the instance $(C_n, \ell, \varphi)$ of the problem Contraction is feasible if and only if every subpath $P \subseteq C_n$ of length $d := |E(P)| \in$

$\{1, 2, \ldots, n-1\}$ satisfies the condition

$$|E(P) \cap C| \leq \lfloor d - \min\{d, n-d\}/\alpha + \beta \rfloor. \tag{3}$$

Rounding down on the right-hand side of (3) is justified because $|E(P) \cap C|$ is always an integer.

Defining

$$\lambda' := \min_{d \in \{1,2,\ldots,n-1\}} \frac{\lfloor d - \min\{d, n-d\}/\alpha + \beta \rfloor}{d}, \tag{4a}$$

$$\lambda := \min\{1, \lambda'\}, \tag{4b}$$

we obtain from (3) that $\lambda \in [0, 1]$ is the maximal amount by which we can contract each edge in a uniform *fractional* solution. Inspired by the rounding technique from [6], we turn this fractional solution into an integer optimal solution, yielding the following greedy algorithm Greedy$(C_n, \alpha, \beta)$: The algorithm considers the edges $e_1, e_2, \ldots, e_n$ of $C_n$ as they are encountered when walking around the cycle. It iteratively constructs a solution $C$ by initializing $C := \emptyset$ and by adding the edge $e_i$ to $C$ if and only if $\lfloor \lambda i \rfloor - \lfloor \lambda(i-1) \rfloor = 1$ for all $i = 1, 2, \ldots, n$ (since $\lambda \in [0, 1]$, this difference is always either 0 or 1). Note that we contract all edges of $C_n$ if and only if $\lambda = 1$.

Theorem 2. *Let $C_n$ be a cycle with unit length edges $\ell = 1$ and consider the tolerance function $\varphi(x) = x/\alpha - \beta$, $\alpha \geq 1$, $\beta \geq 0$. The set of edges computed by the algorithm* Greedy$(C_n, \alpha, \beta)$ *is an optimal solution for the instance $(C_n, \ell, \varphi)$ of the problem* Contraction, *and it is computed in time $\mathcal{O}(n)$.*

The next lemma shows that the contraction computed by our algorithm has the maximum size.

Lemma 3.1. *For any feasible solution $C \subseteq E(C_n)$ we have $|C| \leq \lfloor \lambda n \rfloor$ with $\lambda$ defined in* (4).

*Proof.* If $\lambda = 1$ this inequality is trivial. So let us assume that $\lambda = \lambda' < 1$ and that the minimum in (4a) is attained for some $d \in \{1, 2, \ldots, n-1\}$. Starting at some vertex $u$ of the cycle, we walk along the cycle and cover it with $n$ consecutive paths $P_1, P_2, \ldots, P_n$ of length $d$ each ($P_{i+1}$ starts where $P_i$ ends). The sum of the lengths of the paths is $nd$, so this process ends at the starting vertex $u$, and each edge of the cycle and each edge of $C$ is covered exactly $d$ times. We therefore obtain

$$|C| = \frac{1}{d} \sum_{i=1}^{n} |E(P_i) \cap C| \overset{(3)}{\leq} \sum_{i=1}^{n} \frac{\lfloor d - \min\{d, n-d\}/\alpha + \beta \rfloor}{d} \overset{(4)}{=} \lambda n.$$

As $|C|$ must be integral this inequality yields the desired bound $|C| \leq \lfloor \lambda n \rfloor$. □

With Lemma 3.1 in hand, we are now ready to prove Theorem 2.

*Proof of Theorem* 2. In this proof we will use that for any two real numbers $x$ and $y$ we have

$$\lfloor x \rfloor + \lfloor y \rfloor \leq \lfloor x + y \rfloor, \tag{5a}$$

$$\lfloor x \rfloor - \lfloor y \rfloor \leq \lceil x - y \rceil. \tag{5b}$$

Let $C \subseteq E(C_n)$ be the set of edges computed by the algorithm Greedy$(C_n, \alpha, \beta)$. Clearly, we have $|C| = \sum_{i=1}^{n} (\lfloor \lambda i \rfloor - \lfloor \lambda(i-1) \rfloor) = \lfloor \lambda n \rfloor$, which is optimal by Lemma 3.1.

However, it remains to show that $C$ is feasible. We consider a path $P$ of length $d := |E(P)| \in \{1, 2, \ldots, n{-}1\}$ on the edges $e_k, e_{k+1}, \ldots, e_{k+d-1}$ (indices are considered cyclically modulo $n$, so $e_{n+i} = e_i$). We distinguish two cases: If $k + d - 1 \leq n$, we have

$$|E(P) \cap C| = \sum_{i=k}^{k+d-1} (\lfloor \lambda i \rfloor - \lfloor \lambda(i-1) \rfloor) = \lfloor \lambda(k+d-1) \rfloor - \lfloor \lambda(k-1) \rfloor \overset{(5b)}{\leq} \lceil \lambda d \rceil. \tag{6}$$

If $k + d - 1 > n$, we obtain

$$\begin{aligned} |E(P) \cap C| &= \sum_{i=k}^{n} (\lfloor \lambda i \rfloor - \lfloor \lambda(i-1) \rfloor) + \sum_{i=1}^{d-n+k-1} (\lfloor \lambda i \rfloor - \lfloor \lambda(i-1) \rfloor) \\ &= \lfloor \lambda n \rfloor - \lfloor \lambda(k-1) \rfloor + \lfloor \lambda(d-n+k-1) \rfloor \\ &\overset{(5a)}{\leq} \lfloor \lambda(d+k-1) \rfloor - \lfloor \lambda(k-1) \rfloor \overset{(5b)}{\leq} \lceil \lambda d \rceil. \qquad (7) \end{aligned}$$

Applying (4) and using that $\lceil \lfloor x \rfloor \rceil = \lfloor x \rfloor$ shows that the right-hand sides of (6) and (7) can both be bounded from above by $\lfloor d - \min\{d, n-d\}/\alpha + \beta \rfloor$, proving that $C$ is indeed feasible by (3). ∎

## 4. Dynamic programs for general trees.

In this section we describe dynamic programming algorithms for the problems Contraction and Weak Contraction on trees with general edge lengths and affine tolerance functions. Recall that on trees our objective function satisfies $\Phi(C) = |C|$ for any contraction $C$.

### 4.1. Contraction on trees.

In this section we describe a dynamic programming algorithm for the problem of computing an optimal contraction of a tree $T$ with arbitrary edge lengths $\ell\colon E \to \mathbb{R}_{>0}$ and an affine tolerance function $\varphi(x) = x/\alpha - \beta$, $\alpha \geq 1$, $\beta \geq 0$. The goal is to prove the following result.

Theorem 3. *Let $T$ be a tree with edge lengths $\ell\colon E \to \mathbb{R}_{>0}$ and consider the tolerance function $\varphi(x) = x/\alpha - \beta$, $\alpha \geq 1$, $\beta \geq 0$. An optimal solution for the instance $(T, \ell, \varphi)$ of the problem* Contraction *can be computed by dynamic programming in time $\mathcal{O}(n^3)$.*

Observe that a solution $C \subseteq E$ is feasible if and only if for any two vertices $u$ and $v$ of $T$ we have $\mathrm{load}_{C,\alpha}(u, v) \leq \beta$, where the *load between $u$ and $v$* is defined as

$$\mathrm{load}_{C,\alpha}(u, v) := \mathrm{dist}_\ell(u, v)/\alpha - \mathrm{dist}_{\ell_C}(u, v) \tag{8a}$$

(recall (1)). For any vertex $v$ of $T$ we also define the *load of $T$ at $v$* as

$$\mathrm{load}_{C,\alpha}(T, v) := \max\{\mathrm{load}_{C,\alpha}(u, v) : u \in V(T)\}. \tag{8b}$$

Note that $\mathrm{load}_{C,\alpha}(T, v) \geq 0$, as we have $\mathrm{load}_{C,\alpha}(v, v) = 0$. The next lemma states a criterion when feasible solutions of subtrees can be combined into a feasible solution of the entire tree. The definitions (8a), (8b) and the lemma are illustrated in Figure 2.

Lemma 4.1. *Consider a partition of $T$ into two subtrees $T_1$ and $T_2$ that only have a vertex $v \in V$ in common. Then $C \subseteq E$ is a feasible solution for the instance $(T, \ell, \varphi)$ of the problem* Contraction *if and only if the following two conditions hold: $C \cap E(T_1)$ and $C \cap E(T_2)$ are feasible solutions for the instances $(T_1, \ell, \varphi)$ and $(T_2, \ell, \varphi)$, respectively; and we have $\mathrm{load}_{C,\alpha}(T_1, v) + \mathrm{load}_{C,\alpha}(T_2, v) \leq \beta$.*

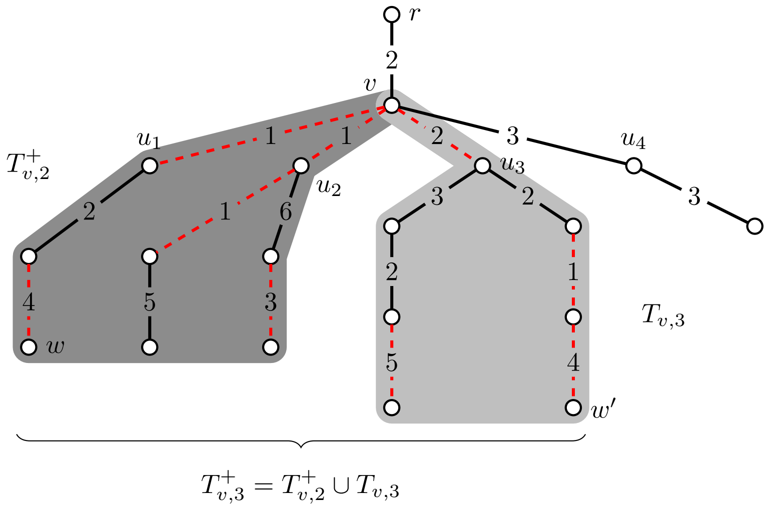

FIG. 2. *Illustration of the definitions of section* 4.1 *for* $(\alpha, \beta) = (2, 4)$. *For the set* $C$ *of dashed edges we obtain* $\mathrm{load}_{C,\alpha}(T^+_{v,2}, v) = \mathrm{load}_{C,\alpha}(v, w) = 7/2 - 2 = 3/2$ *and* $\mathrm{load}_{C,\alpha}(T_{v,3}, v) = \mathrm{load}_{C,2}(v, w') = 9/2 - 2 = 5/2$. *We therefore have* $\mathrm{load}_{C,\alpha}(T^+_{v,2}, v) + \mathrm{load}_{C,\alpha}(T_{v,3}, v) = 4 \leq \beta$, *so by Lemma* 4.1 *the set* $C$ *is a feasible* $(\alpha, \beta)$*-contraction in* $T^+_{v,3}$.

*Proof.* Observe that the path between two vertices $u \in T_1$ and $w \in T_2$ contains the vertex $v$, so we obtain $\mathrm{load}_{C,\alpha}(u, w) = \mathrm{load}_{C,\alpha}(u, v) + \mathrm{load}_{C,\alpha}(v, w)$ from (8a). Using (8b) it follows that the condition $\mathrm{load}_{C,\alpha}(u, w) \leq \beta$ holding for all such pairs of vertices $u, w$ is equivalent to $\mathrm{load}_{C,\alpha}(T_1, v) + \mathrm{load}_{C,\alpha}(T_2, v) \leq \beta$. $\square$

We will use this lemma to formulate our dynamic programming algorithm. The idea is to compute optimal solutions for subtrees and combining them into an optimal solution for the entire tree.

To describe the algorithm we introduce a few definitions. An *ordered rooted tree* is a rooted tree with a specified left-to-right ordering for the children of each vertex. Given the tree $T$, we can pick an arbitrary vertex as the root, and for each descendant of the root an arbitrary left-to-right ordering of its children, yielding an ordered rooted tree (different roots and orderings yield different ordered rooted trees, but any one of them is good for our purposes). We slightly abuse notation in the following and use $T$ to denote this ordered rooted tree. All trees considered in the rest of this section are ordered and rooted. For any vertex $v$ of $T$, we let $T_v$ denote the subtree of $T$ rooted at $v$, and we use $c(v)$ to denote the number of children of $v$. If $u_1, u_2, \ldots, u_{c(v)}$ are the children of $v$ (in the specified ordering), we write $T_{v,i}$, $i \in \{1, \ldots, c(v)\}$, for the subtree of $T$ that contains $v$, $u_i$, and all the descendants of $u_i$. We also define $T_{v,0} := \{v\}$. Furthermore, we define $T^+_{v,i} := \bigcup_{0 \leq j \leq i} T_{v,i}$, so we have $T_v = T^+_{v,c(v)}$. These definitions are illustrated in Figure 2.

Using these definitions it follows straightforwardly from (8a) and (8b) that for any set of edges $C \subseteq E(T_{u_i})$ we have

$$\mathrm{load}_{C \cup \{\{v,u_i\}\},\alpha}(T_{v,i}, v) = \mathrm{load}_{C,\alpha}(T_{u_i}, u_i) + \ell(v, u_i)/\alpha, \tag{9a}$$

$$\mathrm{load}_{C,\alpha}(T_{v,i}, v) = \max\{\mathrm{load}_{C,\alpha}(T_{u_i}, u_i) - (1 - 1/\alpha)\ell(v, u_i), 0\}. \tag{9b}$$

Note that the load increases if the edge $\{v, u_i\}$ is added to $C$ (see (9a)), and it decreases otherwise (see (9b)). Moreover, for any set of edges $C \subseteq T^+_{v,i}$ and any $i = 1, 2, \ldots, c(v)$

we obtain from those definitions that

$$\operatorname{load}_{C,\alpha}(T^+_{v,i}, v) = \max\{\operatorname{load}_{C,\alpha}(T^+_{v,i-1}, v), \operatorname{load}_{C,\alpha}(T_{v,i}, v)\}. \tag{10}$$

These rules allow us to compute the load of all subtrees of $T$ in a bottom-up fashion. Our dynamic program maintains the minimum load of all subtrees of $T$ in three-dimensional matrices $L$ and $L^+$. We begin defining these matrices in an abstract way, and then establish several recursive relations which directly translate into a dynamic program. Specifically, for $v \in V$, $i \in \{0, 1, \ldots, c(v)\}$, and $s \in \{0, 1, \ldots, m\}$ (recall that $m = |E|$) we define

$$L(v, i, s) := \min\{\operatorname{load}_{C,\alpha}(T_{v,i}, v) : C \text{ feasible solution of } (T_{v,i}, \ell, \varphi) \text{ of size } s\}. \tag{11}$$

If there is no feasible solution of the required size, we have $L(v, i, s) = \infty$. The entries of $L^+(v, i, s)$ are defined analogously to (11) by considering the load of $T^+_{v,i}$ instead of $T_{v,i}$. In words, the entries $L(v, i, s)$ and $L^+(v, i, s)$ describe feasible solutions $C$ of size $s$ of the instances $(T_{v,i}, \ell, \varphi)$ or $(T^+_{v,i}, \ell, \varphi)$, respectively, of the problem Contraction for which the load at the vertex $v$ is as small as possible (the matrices contain the minimum achievable load, not the corresponding set of edges).

Lemma 4.2. *Let $v$ be a vertex of $T$ and let $u_1, u_2, \ldots, u_{c(v)}$ be the children of $v$. Then the matrices $L$ and $L^+$ defined in and directly after (11) satisfy the relations*

$$L(v, i, 0) = L^+(v, i, 0) = 0 \quad \textit{for all } i \in \{0, 1, \ldots, c(v)\}, \tag{12a}$$

$$L(v, 0, s) = L^+(v, 0, s) = \infty \quad \textit{for all } s \in \{1, 2, \ldots, m\}, \tag{12b}$$

$$L(v, i, s) = \begin{cases} \mu & \textit{if } \mu \le \beta, \\ \infty & \textit{otherwise}, \end{cases} \tag{12c}$$

*where*

$$\mu := \min\big\{L^+(u_i, c(u_i), s-1) + \ell(v, u_i)/\alpha,\ \max\{L^+(u_i, c(u_i), s) - (1 - 1/\alpha)\ell(v, u_i), 0\}\big\}$$

*for all $i \in \{1, 2, \ldots, c(v)\}$ and $s \in \{1, 2, \ldots, m\}$.*

*Moreover, we have*

$$\begin{aligned} L^+(v, i, s) = \min\big\{ &\max\{L^+(v, i-1, t), L(v, i, s-t)\} : t \in \{0, 1, \ldots, s\} \textit{ and} \\ &L^+(v, i-1, t) + L(v, i, s-t) \le \beta\big\} \end{aligned} \tag{12d}$$

*for all $i \in \{1, 2, \ldots, c(v)\}$ and $s \in \{1, 2, \ldots, m\}$.*

The most interesting of these recursive relations are of course (12c) and (12d). The relation (12c) captures the two possibilities of either adding the edge $\{v, u_i\}$ or not adding it to a partial solution in the tree $T^+_{u_i, c(u_i)} = T_{u_i}$ to obtain a solution for the tree $T_{v,i}$ (recall (9)). The relation (12d), on the other hand, describes how to distribute $s$ contraction edges in $T^+_{v,i}$ among the two subtrees $T^+_{v,i-1}$ and $T_{v,i}$ ($t$ is the number of edges contracted in the first tree, and $s - t$ the number of edges in the second tree, respectively).

*Proof.* The relations (12a) and (12b) follow immediately from the definitions of the trees $T_{v,i}$ and $T^+_{v,i}$ and from (11). The relation (12c) follows from (9) and (11). The relation (12d) follows from (10) and (11) with the help of Lemma 4.1. □

We are now ready to prove Theorem 3.

*Proof of Theorem* 3. Given the instance $(T, \ell, \varphi)$, we fix an arbitrary root $r$ of $T$ and an arbitrary ordering of the children of each vertex, making $T$ an ordered rooted tree. We then compute the entries of the matrices $L$ and $L^+$ using Lemma 4.2. We first initialize various entries using (12a) and (12b), and compute the remaining entries in a bottom-up fashion moving upwards from the leaves to the root. Specifically, at a vertex $v$ with children $u_1, u_2, \dots, u_{c(v)}$ for which all the entries of $L$ and $L^+$ have already been computed, we first compute $L(v, i, s)$ for all $i \in \{1, 2, \dots, c(v)\}$ and $s \in \{1, 2, \dots, m\}$ using (12c), and then $L^+(v, i, s)$ for all $i \in \{1, 2, \dots, c(v)\}$ and $s \in \{1, 2, \dots, m\}$ using (12d).

Let $s^*$ be the largest $s$ such that $L^+(r, c(r), s) \leq \beta$. From (11) we obtain that $s^*$ is the size of an optimal solution of the instance $(T, \ell, \varphi)$. The corresponding set of edges $C \subseteq E$ can be obtained by keeping track of the arguments for which the minima and maxima in (12c) and (12d) are attained in each step.

Clearly, $L$ and $L^+$ both have $\mathcal{O}(n^2)$ entries, and computing each entry takes time $\mathcal{O}(n)$, so the running time of our dynamic program is $\mathcal{O}(n^3)$. □

**4.2. Weak contraction on trees.** In this section we consider the problem of computing weak contractions for a tree $T$ with affine tolerance function $\varphi(x) = x/\alpha - \beta$. Here, our main result is a dynamic programming algorithm that builds on the algorithmic ideas presented in section 4.1.

THEOREM 4. *Let $T$ be a tree with edge lengths $\ell\colon E \to \mathbb{R}_{>0}$ and consider the tolerance function $\varphi(x) = x/\alpha - \beta$, $\alpha \geq 1$, $\beta \geq 0$. An optimal solution for the instance $(T, \ell, \varphi)$ of the problem* WEAK CONTRACTION *can be computed by dynamic programming in time $\mathcal{O}(n^5)$.*

The proof of Theorem 4 is omitted here, and can be found in the preprint version of [11]. In the rest of this section we only outline the main ideas for the proof.

Designing a dynamic programming algorithm for WEAK CONTRACTION on trees is complicated by the fact that the combinability of solutions on subtrees cannot be captured by the load alone. As we need to keep track of pairs of vertices whose distances remain positive when contracting a set of edges $C \subseteq E$, we define the *weak load* of a rooted tree $T$ at one of its vertices $v$ by

$$\operatorname{wload}_{C,\alpha}(T, v) := \max\{\operatorname{load}_{C,\alpha}(u, v) : u \in V(T) \text{ and } \operatorname{dist}_{\ell_C}(u, v) > 0\},$$

allowing us to formulate the following combinability criterion analogous to Lemma 4.1 from before.

LEMMA 4.3. *Let $T$, $T_1$, $T_2$, and $v$ be as in Lemma* 4.1. *Then $C \subsetneq E$ is a feasible solution for the instance $(T, \ell, \varphi)$ of the problem* WEAK CONTRACTION *if and only if the following two conditions hold: For $i = 1, 2$, either $C$ contains every edge of $T_i$ or $C \cap E(T_i)$ is a feasible solution for the instance $(T_i, \ell, \varphi)$ of* WEAK CONTRACTION*; and we have*

$$\operatorname{load}_{C,\alpha}(T_1, v) + \operatorname{wload}_{C,\alpha}(T_2, v) \leq \beta \quad \textit{and} \quad \operatorname{wload}_{C,\alpha}(T_1, v) + \operatorname{load}_{C,\alpha}(T_2, v) \leq \beta.$$

With this lemma in hand, we proceed similarly by computing sets of solutions on rooted subtrees of $T$ that are optimal with respect to the three parameters: size, load, and weak load. In particular, for any fixed size we compute a Pareto front of solutions of that size, minimizing both load and weak load. The key step for arriving at an efficient algorithm is to prove that these Pareto fronts have polynomial, in fact, even linear, size. This is not clear a priori, as the number of feasible solutions on subtrees can be exponential. Using that the weak load has similar monotonicity properties and recursive formulas as stated in (8), (9), and (10) for the load, we obtain an efficient dynamic program. As our algorithm computes $\mathcal{O}(n^2)$ Pareto fronts of size $\mathcal{O}(n)$ at every vertex, and as we can combine optimal solutions from two such fronts in time $\mathcal{O}(n)$, we get an additional factor of $n^2$ in the running time compared to our first dynamic program.

## 5. Hardness for additive tolerance functions.

In this section we prove that the problems Contraction and Weak Contraction for the tolerance function $\varphi(x) = x - \beta$ (purely additive error) are hard already on cycles (section 5.1 below). We then prove that Contraction with the same tolerance function is hard to approximate for general graphs and for bipartite graphs (section 5.2).

### 5.1. Hardness of contraction and weak contraction.

Recall that we can compute optimal (weak) $(\alpha, \beta)$-contractions in polynomial time on trees (this was shown in section 4.1), and have a linear time algorithm for Contraction on cycles with unit length edges (this was shown in section 3). We now show that the problem with $\alpha = 1$ is NP-hard on cycles with arbitrary edge lengths.

Theorem 5. *For any fixed $\beta > 0$, the problems* Contraction *and* Weak Contraction *with tolerance function $\varphi(x) = x - \beta$, $\beta \geq 0$, are NP-hard on cycles.*

Theorem 5 (where $\beta$ is not part of the input) follows immediately from Theorem 6 below (where $\beta$ is part of the input). The reason is that an instance with $\alpha = 1$ does not change when multiplying all edge lengths and $\beta$ by some constant.

Theorem 6. *The problems* Contraction *and* Weak Contraction *with tolerance function $\varphi(x) = x - \beta$, $\beta \geq 0$, are NP-hard on cycles.*

The rest of this section is devoted to proving Theorem 6.

For our proof we will use the following variant of the well-known problem Partition, referred to as Close-to-1 Partition. To state the problem we say that a set of positive rational numbers $\{a_1, a_2, \ldots, a_n\}$ is *close to* 1, if $\sum_{i=1}^n a_i = n$ and $\varepsilon := \sum_{i=1}^n |a_i - 1| < 1/5$.

Close-to-1 Pattition
**Input:** A set of positive rational numbers $\{a_1, a_2, \ldots, a_n\}$ that is close to 1.
**Output:** "Yes" if there is a subset $I \subseteq [n]$ such that $\sum_{i \in I} a_i = \sum_{i \in [n] \setminus I} a_i$, "No" otherwise.

Note that for a "Yes"-instance of this problem, the solution $I \subseteq [n]$ must have size $n/2$, so $|I| = |[n] \setminus I| = \sum_{i \in I} a_i = \sum_{i \in [n] \setminus I} a_i = n/2$. In particular, this implies that $n$ is even.

In the classical problem Partition, the input set is not constrained to be close to 1. Partition was shown to be NP-complete already in Karp's seminal paper [26]. The fact that Close-to-1 Partition is also NP-complete follows from a straightforward rescaling argument.

Lemma 5.1. Close-to-1 Partition *is NP-complete.*

*Proof.* Given an instance $\{a_1, a_2, \ldots, a_n\}$ of Partition, we first add $n$ additional zeroes $a_{n+1} = a_{n+2} = \cdots = a_{2n} = 0$ to the instance (by this we ensure that a partition with equal sums is transformed into one where both partition classes have the same number $n$ of summands). We then linearly transform all the $a_i$ according to $a'_i :=$

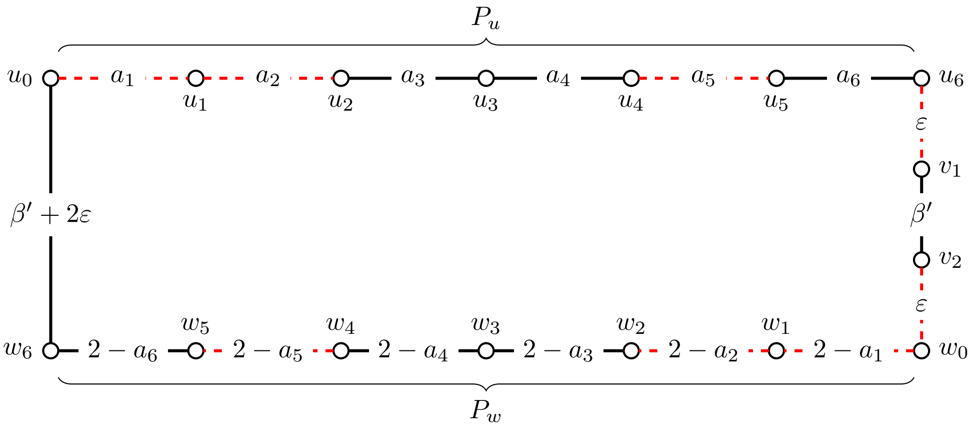

FIG. 3. *Reduction from* CLOSE-TO-1 PARTITION *to* CONTRACTION *for $n = 6$ discussed in the proof of Theorem* 6. *The dashed edges in the figure represent the set $C(I)$ for $I = \{1, 2, 5\}$.*

$(a_i + C)/D$, where $C$ and $D$ are sufficiently large constants so that the transformed values $a_i'$ are close to 1. The transformed set of numbers has even cardinality $2n$, is close to 1, and it admits a partition into two sets of size $n$ with equal sum if and only if the original instance allows a partition into two sets with equal sum. □

*Proof of Theorem* 6. We first focus on the problem CONTRACTION. We reduce CLOSE-TO-1 PARTITION, which is NP-complete by Lemma 5.1, to the problem CONTRACTION on a cycle with tolerance function $\varphi(x) = x - \beta$, $\beta \geq 0$.

Let $\mathcal{I} = \{a_1, a_2, \ldots, a_n\}$ be an instance of CLOSE-TO-1 PARTITION such that $a_1 \geq a_2 \geq \cdots \geq a_n$. This ensures that all $a_i$ that are bigger than 1 appear before all $a_i$ that are smaller than 1, which is the only property of the ordering that we exploit in the proof later on. The instance of CONTRACTION we construct is on the cycle $C_{2n+4}$ with $2n + 4$ edges. We label the vertices of the cycle by walking around the cycle as follows: The first $n + 1$ vertices are labeled $u_0, u_1, \ldots, u_n$, then there are two special vertices $v_1$, $v_2$, and the remaining $n+1$ vertices are labeled $w_0, w_1, \ldots, w_n$; see Figure 3. We denote the subpath $(u_0, \ldots, u_n)$ as $P_u$, and the subpath $(w_0, \ldots, w_n)$ by $P_w$.

We now define $\varepsilon := \sum_{i=1}^{n} |1-a_i| < 1/5$, $\beta := n/2+2\varepsilon$, and $\beta' := \beta+1 > \beta$, and the length function $\ell$ on the cycle edges by setting $\ell(u_{i-1}, u_i) := a_i$ and $\ell(w_{i-1}, w_i) = 2-a_i$ for all $i \in [n]$, and by $\ell(u_n, v_1) = \ell(v_2, w_1) := \varepsilon$, $\ell(v_1, v_2) := \beta'$, and $\ell(w_n, u_0) := \beta'+2\varepsilon$ (see Figure 3).

Now consider the instance $\mathcal{J} := (C_{2n+4}, \ell, \varphi)$ with $\varphi(x) = x - \beta$ of the problem CONTRACTION. Observe that no $\varphi$-contraction may contain an edge $\{u, v\}$ of length greater than $\beta$ (in particular, no feasible solution may contain one of the edges of length $\beta'$ or $\beta' + 2\varepsilon$). Furthermore any (weak) $\varphi$-contraction $C$ on this graph satisfies $\Phi(C) = |C|$.

We will show that $\mathcal{J}$ has an optimal solution of cardinality (and thus of value) $n+2$ if and only if $\mathcal{I}$ is a Yes-instance. In particular, we will see that any feasible solution of $\mathcal{J}$ of size $n + 2$ contains the two edges of length $\varepsilon$ and exactly $n/2$ edges with length $a_i$, $i \in I$, from $P_u$ and the corresponding edges with length $2 - a_i$, $i \in I$, from $P_w$. Such solutions correspond to subsets of $[n]$ in the following natural way: For any subset $I \subseteq [n]$ of size $n/2$ we let $C(I)$ be the subset of edges of the cycle $C_{2n+4}$ consisting of the two edges of length $\varepsilon$ and of all edges $\{u_{i-1}, u_i\}$ and $\{w_{i-1}, w_i\}$ (of length $a_i$ or $2-a_i$, respectively) for all $i \in I$. Thus we will show that $C(I)$ is an optimal solution of the instance $\mathcal{J}$ of Contraction if and only if $\sum_{i\in I} a_i = \sum_{i\in[n]\setminus I} a_i = n/2$, i.e., $\mathcal{I}$ is a Yes-instance of Close-to-1 Partition.

Both directions of this equivalence are captured and proved as Claims 2 and 4 below. Claims 1 and 3 are auxiliary statements used in the proofs of these two main claims.

For any path $P$ on the cycle we let $\ell(P)$ denote the sum of $\ell(e)$ over all edges $e$ of $P$. For all $i \in [n]$ we denote by $P_i^{\sqsupset}$ and $P_i^{\sqsubset}$ the path on the cycle between the vertices $u_i$ and $w_i$ that contains and that does not contain the edge $\{v_1, v_2\}$, respectively (in Figure 3, these are the right and left segment of the cycle).

Claim 1. *For all $i \in [n]$, the number $\ell(P_i^{\sqsupset})$ lies in the interval*

$$[n + \beta' + \varepsilon, n + \beta' + 2\varepsilon]$$

*and the number $\ell(P_i^{\sqsubset})$ lies in the interval $[n + \beta' + 2\varepsilon, n + \beta' + 3\varepsilon]$. In particular, we have $\mathrm{dist}_\ell(u_i, w_i) = \min\{\ell(P_i^{\sqsupset}), \ell(P_i^{\sqsubset})\} = \ell(P_i^{\sqsupset})$ and the difference $\ell(P_i^{\sqsubset}) - \ell(P_i^{\sqsupset})$ lies in the interval $[0, 2\varepsilon]$.*

*Proof of Claim* 1. Note that the condition $\sum_{i=1}^{n} a_i = n$ implies that

$$\varepsilon = 2 \sum_{a_i : a_i \geq 1} (a_i - 1) = 2 \sum_{a_i : a_i < 1} (1 - a_i). \tag{13}$$

By our assumption $a_1 \geq a_2 \geq \cdots \geq a_n$, the numbers $\ell(P_i^{\sqsupset})$ form a unimodal sequence for $i = 0, 1, \ldots, n$ that is maximized for $i = 0$ and $i = n$, proving that $\ell(P_i^{\sqsupset}) \leq n + \beta' + 2\varepsilon$ (note that $\ell(P_u) = \ell(P_w) = n$). By (13) the minimum of this unimodal sequence is at most $\varepsilon$ smaller than the maximum. This proves the first part of the claim. As $\ell(P_i^{\sqsupset}) + \ell(P_i^{\sqsubset}) = 2(n + \beta' + 2\varepsilon)$, we obtain the second part of the claim. The last part of the claim is an immediate consequence of the first two. □

Claim 2. *If $I \subseteq [n]$ is a solution of the instance $\mathcal{I}$ of* Close-to-1 Partition *such that $\sum_{i\in I} a_i = \sum_{i\in[n]\setminus I} a_i = n/2$, then $C(I)$ is a $(1, \beta)$-contraction.*

*Proof of Claim* 2. It suffices to prove that there is no pair of vertices whose distance decreases by more than $\beta$ when contracting the edges in $C(I)$.

We start by verifying this for the pairs $u_i, w_i$ for $i \in [n]$. We first consider the path $P_i^{\sqsupset}$ between $u_i$ and $w_i$. Observe that $\sum_{e\in C(I)\cap P_i^{\sqsupset}} \ell(e)$ lies in the interval $[n/2 + \varepsilon, n/2 + 2\varepsilon] = [\beta - \varepsilon, \beta]$. Similarly to before, this follows from the observation that by the assumption $a_1 \geq a_2 \geq \cdots \geq a_n$ those sums form a unimodal sequence for $i = 0, 1, \ldots, n$ that is maximized for $i = 0$ and $i = n$, and by using (13) (recall also that $|I| = n/2$). Consequently, we have

$$\ell_{C(I)}(P_i^{\sqsupset}) \geq \ell(P_i^{\sqsupset}) - \beta. \tag{14}$$

Since $\sum_{e\in C(I)} \ell(e) = n + 2\varepsilon = 2\beta - 2\varepsilon$, we obtain that $\sum_{e\in C(I)\cap P_i^{\sqsubset}} \ell(e)$ lies in the interval $[\beta - 2\varepsilon, \beta - \varepsilon]$, yielding

$$\ell_{C(I)}(P_i^{\sqsubset}) \geq \ell(P_i^{\sqsubset}) - (\beta - \varepsilon) \geq \ell(P_i^{\sqsubset}) - \beta. \tag{15}$$

Combining (14) and (15) proves that

$$\mathrm{dist}_{\ell_{C(I)}}(u_i, w_i) \geq \mathrm{dist}_\ell(u_i, w_i) - \beta. \tag{16}$$

Now consider two vertices $u_i$ and $w_j$, $j < i$ (the case $j > i$ can be treated analogously). Let $P_{i,j}^{\sqsupset}$ and $P_{i,j}^{\sqsubset}$ be the path on the cycle between the vertices $u_i$ and $w_j$ that contains and that does not contain the edge $\{v_1, v_2\}$, respectively. Using that $P_{i,j}^{\sqsupset} \subseteq P_i^{\sqsupset}$ we obtain

$$\ell_{C(I)}(P_{i,j}^{\sqsupset}) \geq \ell(P_{i,j}^{\sqsupset}) - \beta \tag{17}$$

from (14).

We know that $a_i \leq 1 + 1/5 \leq 8/5$ and consequently

$$2 - a_i \geq 2/5 \geq 2\varepsilon \tag{18}$$

by the assumption that the input $\{a_1, a_2, \dots, a_n\}$ of the instance $\mathcal{I}$ is close to 1 (there is plenty of leeway in all those inequalities). Furthermore, we have

$$\operatorname{dist}_\ell(u_i, w_j) \leq \ell(P_i^{\sqsupset}) - (2 - a_i) \overset{(18)}{\leq} \ell(P_i^{\sqsupset}) - 2\varepsilon \leq \min\{\ell(P_i^{\sqsupset}), \ell(P_i^{\sqsubset})\} = \operatorname{dist}_\ell(u_i, w_i), \tag{19}$$

where the second-to-last inequality follows from Claim 1.

Combining those observations yields

$$\ell_{C(I)}(P_{i,j}^{\sqsubset}) \geq \operatorname{dist}_{\ell_{C(I)}}(u_i, w_i) \overset{(16)}{\geq} \operatorname{dist}_\ell(u_i, w_i) - \beta \overset{(19)}{\geq} \operatorname{dist}_\ell(u_i, w_j) - \beta. \tag{20}$$

Combining (17) and (20) proves that

$$\operatorname{dist}_{\ell_{C(I)}}(u_i, w_j) \geq \operatorname{dist}_\ell(u_i, w_j) - \beta. \tag{21}$$

From (17) and (20) we can derive analogous relations for the remaining cases where we need to consider the distance between a vertex $u_i$, $i \in [n]$, and a vertex $w \in \{v_1, v_2, u_0, u_1, \dots, u_{i-1}, u_{i+1}, \dots, u_n\}$, between a vertex $w_i$, $i \in [n]$, and a vertex $u \in \{v_1, v_2, w_0, w_1, \dots, w_{i-1}, w_{i+1}, \dots, w_n\}$, and between the vertices $v_1$ and $v_2$. This completes the proof of Claim 2. □

Claim 3. *Every $(1, \beta)$-contraction $C$ contains at most $n/2$ edges in $(P_u \cup P_w) \cap P_i^{\sqsupset}$ for all $i \in [n]$ and at most $n/2$ edges in $(P_u \cup P_w) \cap P_i^{\sqsubset}$ for all $i \in [n]$.*

*Proof of Claim* 3. Note that for any $I \subseteq [n]$ and $k \in \{0, 1, \dots, n\}$ we have $\sum_{i \in I : i > k} a_i + \sum_{i \in I : i \leq k} (2 - a_i) \geq |I| - \varepsilon$ by the definition of $\varepsilon$. Consequently, assuming for the sake of contradiction that $C$ contains strictly more than $n/2$ edges in $(P_u \cup P_w) \cap P_i^{\sqsupset}$, we have $\ell_C(P_i^{\sqsupset}) - \ell(P_i^{\sqsupset}) \geq n/2 + 1 - \varepsilon$. Similarly, assuming that $C$ contains strictly more than $n/2$ edges in $(P_u \cup P_w) \cap P_i^{\sqsubset}$ yields $\ell_C(P_i^{\sqsubset}) - \ell(P_i^{\sqsubset}) \geq n/2 + 1 - \varepsilon$. By Claim 1 the difference $\ell(P_i^{\sqsubset}) - \ell(P_i^{\sqsupset})$ lies in the interval $[0, 2\varepsilon]$, so in both cases we obtain

$$\operatorname{dist}_\ell(u_i, w_i) - \operatorname{dist}_{\ell_C}(u_i, w_i) \geq \frac{n}{2} + 1 - \varepsilon - 2\varepsilon > \frac{n}{2} + 2\varepsilon = \beta,$$

where we used that $\varepsilon < 1/5$ in the second-to-last step. This contradicts the fact that $C$ is a $(1, \beta)$-contraction, proving Claim 3. □

Claim 4. *Let $C$ be a feasible solution of the instance $\mathcal{J}$ of* Contraction. *Then we have $|C| \leq n + 2$, and if $|C| = n + 2$, we have $C = C(I)$ for some set $I \subseteq [n]$ with $\sum_{i \in I} a_i = \sum_{i \in [n] \setminus I} a_i = n/2$.*

*Proof of Claim* 4. As $C$ does not contain any of the edges of length $\beta'$ or $\beta' + 2\varepsilon$, we have $|C| \leq n + 2$ by Claim 3 (the $+2$ comes from the two edges of length $\varepsilon$ that may be contained in $C$). Suppose now that $|C| = n + 2$. Applying Claim 3 again shows that $C$ must contain both edges of length $\varepsilon$, and that it contains the edge $\{u_{i-1}, u_i\}$ if and only if it contains the edge $\{w_{i-1}, w_i\}$ for all $i \in [n]$. Defining $I := \{i \in [n] : \{u_{i-1}, u_i\} \in C\}$ we have $|I| = n/2$ and $C = C(I)$.

By Claim 1 we have $\operatorname{dist}_\ell(u_0, w_0) = \ell(P_0^{\sqsupset})$ and $\operatorname{dist}_\ell(u_n, w_n) = \ell(P_n^{\sqsupset})$. As $C$ is a $(1, \beta)$-contraction containing the two edges of length $\varepsilon$ we thus obtain $\sum_{i \in I} a_i = \sum_{e \in C \cap P_u} \ell(e) \leq \beta - 2\varepsilon = n/2$. Similarly, we have $\sum_{i \in [n] \setminus I} a_i = \sum_{i \in I}(2 - a_i) = \sum_{e \in C \cap P_w} \ell(e) \leq \beta - 2\varepsilon = n/2$. As $\sum_{i \in [n]} a_i = n$, these two inequalities must be tight, yielding $\sum_{i \in I} a_i = \sum_{i \in [n] \setminus I} a_i = n/2$. □

Combining Claims 2 and 4 proves the statement of the theorem for the problem CONTRACTION.

We now focus on the problem WEAK CONTRACTION. The hardness result follows immediately from the following claim.

CLAIM 5. *For $n \geq 5$, any feasible weak $(1, \beta)$-contraction $C$ on the instance $\mathcal{J}$ is also a feasible $(1, \beta)$-contraction.*

*Proof of Claim* 5. Suppose for the sake of contradiction that $C$ is not a feasible $(1, \beta)$-contraction. This means there are vertices $a, b$ such that $\operatorname{dist}_{\ell_C}(a, b) = 0$ and $\operatorname{dist}_\ell(a, b) > \beta$, i.e., $a$ and $b$ lie on a (maximal) subpath $Q$ formed by edges from $C$ on the cycle. Let $u$ be one end vertex of $Q$, and let $x$ be the neighbor of $u$ not on $Q$. Let $v$ be the last vertex on $Q$ when traversed starting at $u$, such that the length of the $x$-$v$-path $P$ containing $u$ is at most $\beta + \ell(x, u)$, and let $y$ be the next vertex on $Q$ when traversed starting at $u$. Such a vertex $y$ exists as $\ell(Q) > \beta$, and the $x$-$y$-path $P'$ containing $u$ has length strictly greater than $\beta + \ell(x, u)$.

We have $\operatorname{dist}_{\ell_C}(x, y) > 0$, as $C$ does not contract the entire cycle. By (1), we have $\operatorname{dist}_{\ell_C}(x, y) \geq \operatorname{dist}_\ell(x, y) - \beta$. As $\operatorname{dist}_{\ell_C}(x, y) \leq \ell(x, u)$, we get $\operatorname{dist}_\ell(x, y) \leq \beta + \ell(x, u)$. As we saw before, the $x$-$y$-path $P'$ has length strictly greater than $\beta + \ell(x, u)$, thus the $x$-$y$-path $P''$ not containing $u$ must have length at most $\beta + \ell(x, u)$. As the entire cycle has length $2n + 2\beta' + 4\varepsilon = 3n + 2 + 8\varepsilon$ and can be partitioned into $P, P''$, and the edge $\{v, y\}$, we get

$$\begin{aligned} 3n + 2 + 8\varepsilon &= \ell(P) + \ell(P'') + \ell(v, y) \\ &\leq 2(\beta + \ell(x, u)) + \ell(v, y) \leq 5\beta + 3 + 4\varepsilon = 5n/2 + 3 + 14\varepsilon, \end{aligned}$$

where the second inequality holds as the two longest edges of the cycle have length $\beta' + 2\varepsilon = \beta + 1 + 2\varepsilon$ and $\beta' = \beta + 1$, respectively. From this chain of inequalities we obtain $n \leq 2 + 12\varepsilon < 4 + 2/5$, contradicting the assumption $n \geq 5$. □

This ends the proof of Theorem 6. □

The reader might be tempted to "simplify" the previous reduction proof by omitting the four special edges of length $\varepsilon$, $\beta'$, and $\beta' + 2\varepsilon$ and by setting $\beta := n/2$ instead. However, this would invalidate Claim 2 (specifically, the estimate (15) would not always hold).

**5.2. Inapproximability of contraction.** We are able to extend the aforementioned hardness result for CONTRACTION as follows:

THEOREM 7. *For any fixed $\beta > 0$ and $\varepsilon > 0$, it is NP-hard to approximate the problem* CONTRACTION *with tolerance function $\varphi(x) = x - \beta$, $\beta \geq 0$, to within a factor of $n^{1-\varepsilon}$.*

For the following theorem the additive error is fixed to $\beta = 1$.

THEOREM 8. *For any $\varepsilon > 0$, it is NP-hard to approximate the problem* CONTRACTION *with tolerance function $\varphi(x) = x - 1$ on bipartite graphs with unit length edges $\ell = 1$ to within a factor of $m^{1/2-\varepsilon}$.*

Our reductions are based on the inapproximability of the well-known CLIQUE problem. Recall that a *clique* in a graph $G$ is a complete subgraph of $G$.

CLIQUE
**Input:** A graph $G$.
**Output:** A clique in $G$ of maximum size.

It was shown in [36] that for any $\varepsilon > 0$, it is NP-hard to approximate CLIQUE to within a factor of $n^{1-\varepsilon}$.

The following lemma will be used in our proofs. It shows that for $(1, \beta)$-contractions the feasibility condition (1) need not be checked for all pairs of vertices $u$ and $v$, but only for those satisfying certain extra conditions.

LEMMA 5.2. *A set of edges $C \subseteq E$ is a $(1, \beta)$-contraction if and only if all pairs of vertices $u, v \in V$ with the property that every shortest path with respect to $\ell_C$ between $u$ and $v$ starts and ends with an edge from $C$ satisfying condition* (1).

*Proof.* Suppose for the sake of contradiction that all pairs of vertices $u, v \in V$ as in the lemma satisfy condition (1) and that $C$ is *not* a $(1, \beta)$-contraction. Then there is a pair of vertices $u, v \in V$ violating (1) and a shortest path $P$ with respect to $\ell_C$ between $u$ and $v$ that does not start or end with an edge from $C$. We choose $u$ and $v$ such that $\mathrm{dist}_{\ell_C}(u, v)$ is minimal, and we may assume that the first edge $\{u, w\}$ of $P$ is not contained in $C$, so $\mathrm{dist}_\ell(u, v) - \mathrm{dist}_{\ell_C}(u, v) = \mathrm{dist}_\ell(w, v) - \mathrm{dist}_{\ell_C}(w, v)$. By our choice of $u$ and $v$, the vertices $w$ and $v$ satisfy (1), i.e., the right-hand side of this equation is bounded by $\beta$, a contradiction. □

*Proof of Theorem* 7. Let $\beta, \varepsilon > 0$ be fixed and let $G = (V, E)$ be an instance of CLIQUE.

We define a graph $H = H(G)$ as follows (see Figure 4): The vertex set of $H$ is given by $(V \times \{1, 2\}) \cup \{s\}$, i.e., we create two copies of each original vertex and add a special vertex $s$. The edge set of $H$ is given by $\{\{(u, 1), (v, 1)\} : \{u, v\} \in E\}$ plus the edges $\{(v, 1), (v, 2)\}$ and $\{s, (v, 2)\}$ for all $v \in V$. The first set of edges are simply the original edges of $G$ on the first copies of the vertices, the second set is a perfect matching between the two copies of the vertex set, and the third set of edges connects the special vertex $s$ to all vertices of the second copy of the vertex set. The length function $\ell$ on the edges of $H$ is set to $2\beta + 2$, $\beta$, or $\beta + 1$ for those three sets of edges, respectively.

Now consider the instance $\mathcal{I} := (H, \ell, \varphi)$ of the problem CONTRACTION with the tolerance function $\varphi(x) = x - \beta$. Clearly, any $(1, \beta)$-contraction $C$ in $H$ can contain only edges of the form $\{(u, 1), (u, 2)\}$ for some $u \in V$. As $H$ does not contain two edges between two different connected components of $(V, C)$, our objective function defined in (2) satisfies $\Phi(C) = |C|$ for any feasible solution $C$ of $\mathcal{I}$. We will show that it allows a feasible solution with $k$ edges (and thus of value $k$) if and only if $G$ has a clique with $k$ vertices. Formally, for $U \subseteq V$ we define $C(U) := \{\{(u, 1), (u, 2)\} : u \in U\}$ (see Figure 4). We proceed to show that $U$ induces a clique in $G$ if and only if $C(U)$ is a $(1, \beta)$-contraction in $H = H(G)$.

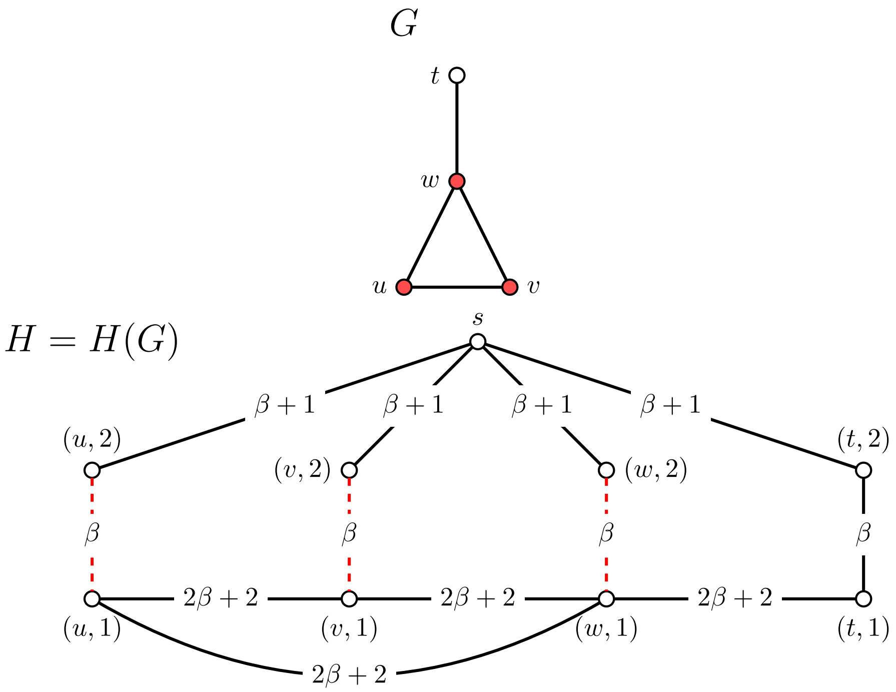

FIG. 4. *An instance $G$ of* CLIQUE *(left) and the corresponding instance $\mathcal{I} = (H, \ell, \varphi)$ (right) of* CONTRACTION *constructed in the proof of Theorem* 7*. The dashed edges form the set $C(U)$ for $U = \{u, v, w\}$.*

Note that for any two vertices $u, v \in U$ we have

$$\begin{aligned}
\operatorname{dist}_{\ell_{C(U)}}((u,1),(v,1)) &= 2\beta + 2 = \begin{cases} \operatorname{dist}_\ell((u,1),(v,1)) & \text{if } \{u,v\} \in E, \\ \operatorname{dist}_\ell((u,1),(v,1)) - 2\beta & \text{otherwise,} \end{cases} \\
\operatorname{dist}_{\ell_{C(U)}}((u,1),(v,2)) &= 2\beta + 2 = \operatorname{dist}_\ell((u,1),(v,2)) - \beta, \\
\operatorname{dist}_{\ell_{C(U)}}((u,2),(v,2)) &= 2\beta + 2 = \operatorname{dist}_\ell((u,2),(v,2)), \\
\operatorname{dist}_{\ell_{C(U)}}((u,1),(u,2)) &= 0 = \operatorname{dist}_\ell((u,1),(u,2)) - \beta.
\end{aligned}$$

These relations together with Lemma 5.2 show that $C(U)$ is a $(1, \beta)$-contraction in $H$ if and only if $U$ is a clique in $G$.

As $n(H)$ differs from $n(G)$ only by a constant factor, an $n^{1-\varepsilon}$-approximation algorithm for CONTRACTION would yield an $n^{1-\varepsilon'}$-approximation algorithm for CLIQUE via this reduction. Together with the aforementioned inapproximability of CLIQUE [36] this proves the theorem. ∎

The rest of this section is devoted to proving Theorem 8, so we now focus on $(1, 1)$-contractions in bipartite graphs with unit length edges $\ell = 1$. The next lemma characterizes the structure of contractions in this setting.

LEMMA 5.3. *Let $G = (V, E)$ be a bipartite graph with unit edge lengths $\ell = 1$ and let $C \subseteq E$ be a set of edges.*

(i) *If $C$ is a $(1, 1)$-contraction, then $C$ is a matching.*

(ii) *If $C = \{e, f\}$ with edges $e = \{u_1, u_2\}, f = \{v_1, v_2\} \in E$, then $C$ is a $(1,1)$-contraction if and only if* $\mathrm{dist}_\ell(u_1, v_1) = \mathrm{dist}_\ell(u_2, v_2)$ *and* $\mathrm{dist}_\ell(u_1, v_2) = \mathrm{dist}_\ell(u_2, v_1)$.

(iii) *$C$ is a $(1,1)$-contraction if and only if all two-element subsets of $C$ are.*

*Proof.*

(i) Suppose for the sake of contradiction that $C$ contains a path $(u, v, w)$ on two edges. As $G$ is bipartite, it has no triangles, so $\mathrm{dist}_\ell(u, w) = 2$ and $\mathrm{dist}_{\ell_C}(u, w) = 0$, a contradiction to the assumption that $C$ is a $(1,1)$-contraction.

(ii) For the edges $e = \{u_1, u_2\}$ and $f = \{v_1, v_2\}$ we define $d_{i,j} := \mathrm{dist}_\ell(u_i, v_j)$ for $i, j \in \{1, 2\}$.
Let $C = \{e, f\}$ be a $(1,1)$-contraction. Both $d_{1,1}$ and $d_{2,2}$ must have the same parity (as $G$ is bipartite), so if $d_{1,1} < d_{2,2}$, the difference between them is exactly 2. However, this would mean that $\mathrm{dist}_{\ell_C}(u_2, v_2) = d_{1,1} = d_{2,2} - 2 = \mathrm{dist}_\ell(u_2, v_2) - 2$, a contradiction to the assumption that $C$ is a $(1,1)$-contraction. Repeating the same argument with $d_{1,1}$ and $d_{2,2}$ interchanged shows that $d_{1,1} = d_{2,2}$. An analogous argument shows that $d_{1,2} = d_{2,1}$.
Now suppose that $d_{1,1} = d_{2,2}$ and $d_{1,2} = d_{2,1}$. From these conditions it follows that for all $i, j \in \{1, 2\}$ every path between $u_i$ and $v_j$ that contains both edges $e$ and $f$ has length at least $d_{i,j} + 2$ with respect to $\ell$. Consequently, we have $\mathrm{dist}_{\ell_C}(u_i, v_j) \geq \mathrm{dist}_\ell(u_i, v_j) - 1$ for $C = \{e, f\}$. By Lemma 5.2, $C$ is a $(1,1)$-contraction.

(iii) One direction of the equivalence is obvious, so we only need to prove the other direction. So we assume that all two-element subsets of $C$ are $(1,1)$-contractions, and we need to prove that $C$ is a $(1,1)$-contraction. The argument is a straightforward generalization of the argument for (ii) from before. Let $P$ be a path that contains exactly $k$ edges from $C$, and that starts and ends with an edge from $C$. Let $e_1, e_2, \ldots, e_k$ be those edges and $u_{1,1}, u_{1,2}, u_{2,1}, u_{2,2}, \ldots, u_{k,1}, u_{k,2}$ their end vertices as they are encountered when traversing $P$ (so $u_{1,1}$ and $u_{k,2}$ are the end vertices of $P$). For all $i = 1, 2, \ldots, \lfloor k/2 \rfloor$ the pair of edges $e_{2i-1}$ and $e_{2i}$ and their end vertices satisfy the distance conditions from (ii). From these conditions it follows that the subpath of $P$ between $u_{2i-1,1}$ and $u_{2i,2}$ has length at least $\mathrm{dist}_\ell(u_{2i-1,1}, u_{2i,2}) + 2$. So overall the length of $P$ is at least $\mathrm{dist}_\ell(u_{1,1}, u_{k,2}) + 2\lfloor k/2 \rfloor \geq \mathrm{dist}_\ell(u_{1,1}, u_{k,2}) + (k-1)$. Consequently, we have $\mathrm{dist}_{\ell_C}(u_{1,1}, u_{k,2}) \geq \mathrm{dist}_\ell(u_{1,1}, u_{k,2}) - 1$. By Lemma 5.2, $C$ is a $(1,1)$-contraction. ∎

With Lemma 5.3 in hand, we are now ready to prove Theorem 8.

*Proof of Theorem* 8. Let $\varepsilon > 0$ be fixed and let $G = (V, E)$ be an instance of Clique. We construct a bipartite graph $H = H(G)$ as follows (see Figure 5): For every vertex $v \in V$, the graph $H$ contains two vertices $(v, 1)$ and $(v, 2)$ and the edge $f_v := \{(v, 1), (v, 2)\}$. For every edge $e = \{u, v\} \in E$, we add a vertex $x_e$ and the edges $f_{e,u} := \{x_e, (u, 1)\}$ and $f_{e,v} := \{x_e, (v, 1)\}$ to $H$. Furthermore, we add a new special vertex $s$ to $H$ and all the edges $\{s, (v, 2)\}$, $v \in V$, and $\{s, x_e\}$, $e \in E$. It is easy to check that the graph $H$ defined in this way is bipartite.

All edges of $H$ receive unit lengths ($\ell = 1$) and we consider the instance $\mathcal{I} = (H, \ell, \varphi)$ of the problem Contraction with the tolerance function $\varphi(x) = x - 1$.

For any set of vertices $U \subseteq V$ we define $C(U) := \{f_u : u \in U\}$ (see Figure 5).

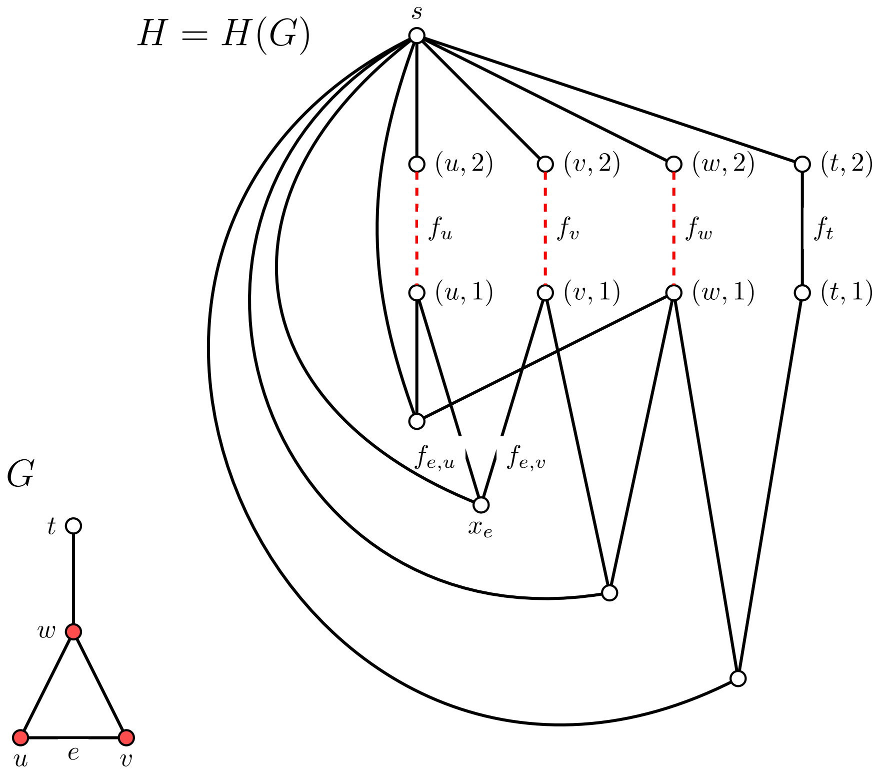

FIG. 5. *An instance $G$ of* CLIQUE *(left) and the corresponding instance $\mathcal{I} = (H, \ell, \varphi)$ (right) of* CONTRACTION *constructed in the proof of Theorem* 8. *The dashed edges form the set $C(U)$ for $U = \{u, v, w\}$.*

CLAIM 6. *If $U \subseteq V$ is a clique in $G$, then $C(U)$ is a $(1,1)$-contraction in $H$ and $\Phi(C(U)) = |U|$.*

*Proof of Claim* 6. Let $U$ be a a set of vertices in $G$ that form a clique, and let $u, v \in U$ be two vertices from this clique. Then we have $\text{dist}_\ell((u,1),(v,1)) = \text{dist}_\ell((u,2),(v,2)) = 2$ and $\text{dist}_\ell((u,1),(v,2)) = \text{dist}_\ell((u,2),(v,1)) = 3$, so Lemma 5.3(ii) implies that $C(\{u,v\})$ is a $(1,1)$-contraction in $H$. Repeating this argument for every pair of vertices from $U$ and applying Lemma 5.3(iii) yields that $C(U)$ is a $(1,1)$-contraction in $H$. As there are never two edges in $H$ between any two connected components of the graph $(V, C(U))$, we have $\Phi(C(U)) = |C(U)| = |U|$. □

For any set of edges $C \subseteq E(H)$, we let $U(C)$ be the set of vertices $v \in V$ for which $(v,1)$ is incident to an edge in $C$.

CLAIM 7. *If $C \subseteq E(H)$ is a $(1,1)$-contraction, then $C$ is a matching in $H$ and $U(C)$ is a clique in $G$ of size at least $\Phi(C) - 3$.*

*Proof of Claim* 7. $C$ is a matching by Lemma 5.3(i).

Let $u, v \in U(C)$. We will show that $e = \{u, v\} \in E$ by applying Lemma 5.3(ii) to the two edges in $C$ incident to $(u,1)$ and $(v,1)$. To prove that $e \in E$ it suffices to show that $\text{dist}_\ell((u,1),(v,1)) = 2$.

Let us first consider the case that $f_u, f_v \in C$. As $\text{dist}_\ell((u,2),(v,2)) = 2$ (the shortest path between those vertices goes via $s$), Lemma 5.3(ii) implies that

$\text{dist}_\ell((u,1),(v,1)) = 2$. We now consider the case that there is an edge $e' \in E \setminus \{e\}$ with $f_u, f_{e',v} \in C$. We then have $\text{dist}_\ell((u,2),x_{e'}) = 2$ (via $s$), so Lemma 5.3(ii) yields $\text{dist}_\ell((u,1),(v,1)) = 2$. Finally, we consider the case that there are two edges $e', e'' \in E \setminus \{e\}$ with $f_{e',u}, f_{e'',v} \in C$. We then have $\text{dist}_\ell(x_{e'}, x_{e''}) = 2$ (via $s$), again implying that $\text{dist}_\ell((u,1),(v,1)) = 2$. This proves that indeed $e \in E$, so $U(C)$ forms a clique in $G$.

Every edge in $H$ is either incident to $s$ or to a vertex of the form $(v,1)$, $v \in V$. Since at most one of the edges incident to $s$ can be in $C$, the definition of $U(C)$ shows that the size of $U(C)$ is either $|C|-1$ or $|C|$. Therefore, to finish the proof of Claim 7, it suffices to show that $\Phi(C) \leq |C|+2$. If $C$ contains no two edges that are connected by more than one edge in $H$, then we have $\Phi(C) = |C|$. Otherwise we consider two such edges $f$ and $g$ from $C$. It is easy to check that either $f$ or $g$ must be incident to $s$, so suppose that the edge $f$ contains $s$. We first consider the case that $f = \{s, x_e\}$ for some edge $e = \{u,v\} \in E$. In this case it follows that $g = \{(u,1),(u,2)\}$ or $g = \{(v,1),(v,2)\}$, so we have $\Phi(C) = |C|+2$. Now consider the case that $f = \{s,(u,2)\}$ for some vertex $u \in V$. In this case it follows that $g = \{(u,1), x_e\}$ for exactly one edge $e \in E$ incident to $u$ in $G$, showing that $\Phi(C) = |C|+2$. In all three cases we have $\Phi(C) \leq |C|+2$, as claimed. ∎

Combining Claims 6 and 7 will allow us to prove the following claim.

CLAIM 8. *If there is an $n^{1/2-\varepsilon}$-approximation algorithm for* CONTRACTION*, then there is an $n^{1-\varepsilon/2}$-approximation algorithm for* CLIQUE.

*Proof of Claim* 8. Suppose for the sake of contradiction that such an approximation algorithm for CONTRACTION exists. We use it to compute a clique in a given instance $G$ of CLIQUE as follows: We construct $\mathcal{I} = (H(G), \ell, \varphi)$ and compute a solution $C$ of CONTRACTION for this instance, and we define the clique $U(C)$ as before (recall Claim 7). If $U(C) \neq \emptyset$, we return $U(C)$, otherwise we return any vertex from $G$. We denote the clique computed in this fashion by $U$.

We may assume that $n(G) \geq 16^{1/\varepsilon}$, in particular, $n(H) \geq 16^{1/\varepsilon}$. It follows that

$$n(H) = 1 + 2n(G) + m(G) \leq 1 + 2n(G) + \binom{n(G)}{2} \leq n(G)^2. \tag{22}$$

By assumption we know that

$$\Phi(C) \cdot n(H)^{1/2-\varepsilon} \geq \Phi(C^*), \tag{23}$$

where $C^*$ is an optimal solution of $\mathcal{I}$. In particular, $\Phi(C)$ is positive.

Combining these observations we get

$$\begin{aligned} |U| \cdot n(G)^{1-\varepsilon/2} &\overset{(22)}{\geq} |U| \cdot n(H)^{1/2-\varepsilon/2} \\ &\geq \max\{\Phi(C)-3, 1\} \cdot n(H)^{1/2-\varepsilon/2} \\ &= \big(\max\{\Phi(C)-3, 1\} \cdot n(H)^{\varepsilon/2}\big) \cdot n(H)^{1/2-\varepsilon} \\ &\geq \Phi(C) \cdot n(H)^{1/2-\varepsilon} \\ &\overset{(23)}{\geq} \Phi(C^*) \\ &\geq \omega(G), \end{aligned}$$

where the second inequality holds because of Claim 7, and the last inequality involving the clique number $\omega(G)$ holds because of Claim 6. ∎

As $m(H) = \Theta(n(H))$, Claim 8 implies the theorem (using the inapproximability of CLIQUE proved in [36]). ◻

## 6. Hardness for multiplicative tolerance function.

By Theorem 5, the problem WEAK CONTRACTION with purely additive tolerance function $\varphi(x) = x - \beta$ is NP-hard on cycles. In this section we prove the hardness and inapproximability of this problem also in the case of a purely multiplicative tolerance function $\varphi(x) = x/\alpha$, $\alpha \geq 1$. Recall that the problem CONTRACTION is trivial for this tolerance function (we may not contract any edges). The proofs of the two main results of this section, Theorems 9 and 10 below, are omitted here and can be found in the preprint version of [11].

To state the next result recall that the *girth* of a graph $G$ is defined as the minimum length of a cycle in $G$.

THEOREM 9. *For any $g \geq 2$, the problem* WEAK CONTRACTION *with tolerance function $\varphi(x) = x/2$ is NP-hard for planar graphs with girth at least $3g$ and unit length edges $\ell = 1$.*

Theorem 9 implies that WEAK CONTRACTION is hard for a general multiplicative tolerance function $\varphi(x) = x/\alpha$, $\alpha \geq 1$, but it leaves open the question of whether this is true also for other fixed values of $\alpha$ other than 2 (when $\alpha$ is not part of the input). The arguments in the proof for $\alpha = 2$ carry over straightforwardly to any fixed value $2 \leq \alpha < 3$, but not to 3 or larger values (for $\alpha < 2$ and unit length edges the problem is trivial).

We prove Theorem 9 by a reduction from a variant of the PLANAR 3SAT problem, which is characterized by additional properties of the bipartite variable-clause graph.

We are able to extend our hardness results for WEAK CONTRACTION as follows.

THEOREM 10. *For any $\varepsilon > 0$, it is NP-hard to approximate the problem* WEAK CONTRACTION *with tolerance function $\varphi(x) = 2x/3$ to within a factor of $n^{1-\varepsilon}$.*

Theorem 10 implies that WEAK CONTRACTION is hard to approximate for general multiplicative tolerance functions $\varphi(x) = x/\alpha$, $\alpha \geq 1$, but it leaves open the question of whether this is true also for other fixed values of $\alpha$ other than $3/2$ (when $\alpha$ is not part of the input). The arguments in the proof for $\alpha = 3/2$ carry over straightforwardly to any fixed value $1 < \alpha < 2$, but not to 2 or larger values (for $\alpha = 1$ the problem is trivial).

We prove Theorem 10 by a reduction from the well-known INDEPENDENT SET problem (which is equivalent to CLIQUE by considering the complement graph).

## 7. Asymptotic bounds.

In this section we show how to compute contractions for graphs that are not optimal, but can be computed efficiently despite our hardness results from the previous section. In this vein, the main results of this section are Theorem 11 and the corresponding (not tight) lower bound (Theorem 13) for the case of tolerance functions of the form $\varphi(x) = x/\alpha - 1$. Further we consider purely additive tolerance functions (section 7.2) and the factor by which a contraction can reduce the number of vertices (section 7.3). Throughout this section, we assume all graphs to have unit length edges $\ell = 1$.

### 7.1. Almost multiplicative contractions.

As mentioned in the introduction, a purely multiplicative tolerance function ($\beta = 0$) forbids decreasing any distances. In this section we thus consider an "almost" purely multiplicative tolerance function of the form $\varphi(x) = x/\alpha - 1$.

Theorem 11. *Let $k \geq 1$ be a real number. Any graph $G$ has a $(2k-1,1)$-contraction $C$ such that the contracted graph $G/C$ has at most $n^{1+1/k}$ edges, and such a contraction can be computed in time $\mathcal{O}(m)$.*

Recall that here and throughout, $n$ and $m$ denote the number of vertices and edges of the input graph $G$, not of the contracted graph $G/C$. Setting $k := \log_2 n$ in Theorem 11 yields the following corollary.

Corollary 12. *Any graph $G$ has a $(2\log_2 n - 1, 1)$-contraction $C$ such that the contracted graph $G/C$ has at most $2n$ edges, and such a contraction can be computed in time $\mathcal{O}(m)$.*

To prove Theorem 11, we use a clustering approach as presented in [4], yielding the next lemma. Specifically, the following crucial lemma appears in a slightly weaker form in that paper. For any real number $r \geq 1$, we define an *$r$-partition* of a graph $G = (V,E)$ as a set of *clusters* $P_i \subseteq V$, $i \in [l]$, with corresponding *cluster centers* $p_i \in P_i$, where the sets $P_i$ are required to form a partition of the vertex set $V$ and where $\mathrm{dist}_\ell(p_i, u) \leq r-1$ for all $u \in P_i$ and $i \in [l]$. We denote the resulting $r$-partition by $P := \{(p_i, P_i) : i \in [l]\}$. We write $\rho(P)$ for the number of pairs $1 \leq i < j \leq l$ for which $P_i$ and $P_j$ are connected by at least one edge, and we refer to this quantity as the *density of $P$*.

Lemma 7.1. *Let $r \geq 1$ be a real number. Any graph $G$ with unit length edges has an $r$-partition $P$ with density $\rho(P) \leq n^{1+1/r}$, and such a partition can be computed in time $\mathcal{O}(m)$.*

*Proof.* The idea of the algorithm is to build an $r$-partition $P$ of $G$ iteratively in rounds. In each round, we build a new cluster and remove all vertices from that cluster from the graph, processing the subgraph on the remaining vertices in the next round. The algorithm proceeds until all vertices are assigned to a cluster. In round $i$, we choose an arbitrary vertex $p_i$ as a cluster center, and define layers $L_{i,0}, L_{i,1}, \ldots$ around the vertex $p_i$, where the layer $L_{i,j}$ consists of all vertices at distance exactly $j$ from $p_i$ (this distance is measured in the subgraph of $G$ under consideration in this round). We continue computing these layers as long as the number of vertices in the new layer is at least the number of vertices in all previous layers times the factor $n^{1/r}$. The cluster $P_i$ is defined as the union of all layers around $p_i$ satisfying this expansion condition. We refer to the first layer violating this condition (which is not added to $P_i$ anymore) as the *rejected layer*. We let $P$ denote the partition of the vertices of $G$ computed in this fashion.

To verify that $P$ is indeed an $r$-partition, we proceed to show that each vertex within a cluster has distance at most $r-1$ from the center vertex of that cluster, and that the density $\rho(P)$ of the partition is at most $n^{1+1/r}$. Intuitively, the expansion condition in the definition of the layers ensures that a cluster has few layers and that the number of edges that go to unclustered vertices is small.

Consider a cluster $P_i$ with center vertex $p_i$ and the layers $L_{i,0}, L_{i,1}, \ldots, L_{i,d}$. Suppose for the sake of contradiction that $d \geq r$. By the definition of the layers in the algorithm we know that $|L_{i,j}| \geq n^{1/r} \sum_{k=0}^{j-1} |L_{i,k}|$ holds for all $j \in [d]$, implying that $|L_{i,j}| \geq n^{j/r}$. Consequently, the size of the cluster satisfies $|P_i| = \sum_{j=0}^{d} |L_{i,j}| \geq 1 + n^{r/r} = n+1$, a contradiction.

We now show that $\rho(P) \leq n^{1+1/r}$. The key idea is that the number of vertices in the rejected layer of a cluster $P_i$ is at most $n^{1/r}|P_i|$. Thus the number of edges from $P_i$ to clusters that are created later is at most $n^{1/r}|P_i|$. For every edge between two clusters we let the cluster that is created first account for that edge. Summing over all these edges between clusters yields the desired upper bound of $\rho(P) \leq n \cdot n^{1/r} = n^{1+1/r}$.

Using breadth-first search, the partitioning algorithm described above runs in time $\mathcal{O}(m)$ (recall that $G$ is assumed to be connected). This completes the proof of the lemma. ◻

With Lemma 7.1 in hand, we are now ready to prove Theorem 11.

*Proof of Theorem* 11. Given $G = (V, E)$, we first compute a $k$-partition $P$ into $l$ clusters as described by Lemma 7.1. We define the set $C$ of contracted edges as the union of all edges within the clusters, $C := \{\{u, v\} \in E : u, v \in P_i \text{ for some } i \in [l]\}$. We thus contract each cluster into a single vertex and remove from every set of resulting parallel edges all but a single edge.

We proceed to show that $C$ is a $(2k-1, 1)$-contraction, i.e., we show that $\mathrm{dist}_{\ell_C}(u, v) \geq \mathrm{dist}_\ell(u, v)/(2k-1) - 1$ for all $u, v \in V$. Consider two vertices $u \in P_i$ and $v \in P_j$, where $i$ and $j$ might be equal. Let $Q_{u,v}$ be the shortest path from $u$ to $v$ in $G$ with edge lengths $\ell_C$ (all edges from $C$ receive length zero). The length $d$ of $Q_{u,v}$ is the number of edges on that path that connect different clusters. Note that $Q_{u,v}$ enters and leaves each of the $d+1$ visited clusters at most once, using at most $2k-2$ edges in every cluster, so in $G$ (where all edges have unit lengths) we get $\mathrm{dist}_\ell(u, v) \leq d + (d+1)(2k-2)$.

Combining these observations we obtain

$$\mathrm{dist}_{\ell_C}(u, v) = d \geq d - \frac{1}{2k-1} = \frac{d + (d+1)(2k-2)}{2k-1} - 1 \geq \frac{\mathrm{dist}_\ell(u, v)}{2k-1} - 1,$$

proving the claim. It remains to show that the contracted graph $G/C$ has at most $n^{1+1/k}$ edges, which is an immediate consequence of the upper bound $m(G/C) = \rho(P) \leq n^{1+1/k}$ given by Lemma 7.1. This completes the proof of the theorem. ◻

Erdős' girth conjecture [23] asserts that there exist graphs with $\Omega(n^{1+1/k})$ edges and girth $2k+1$. It has been verified for $k = 1, 2, 3, 5$ [34] and the strongest spanner lower bounds depend on it. We derive from the conjecture the following (not tight) lower bound.

THEOREM 13. *Assuming Erdős' girth conjecture, there exists for any integer $k \geq 2$ a graph $G$ such that any $(k-1, 1)$-contraction $C$ results in a graph $G/C$ with $\Omega(n^{1+1/k})$ edges.*

*Proof.* For a given integer $k \geq 2$ let $G$ be a graph that is guaranteed by Erdős' girth conjecture, i.e., $G$ has girth $2k+1$ and $\Omega(n^{1+1/k})$ edges. Consider any $(k-1, 1)$-contraction $C$ on $G$, and consider a connected component of the graph $(V, C)$. Applying (1) shows that $\mathrm{dist}_\ell(u, v) \leq k-1$ holds for any two vertices $u$ and $v$ in that component. Using that the girth of $G$ is $2k+1$, it follows that for any cycle in $G$, the connected component of $(V, C)$ does not contain a contiguous segment of cycle edges of length at least half of the cycle. This implies that all connected components of the graph $(V, C)$ are trees with diameter at most $k-1$. Therefore, the total number of edges within all connected components of $(V, C)$ is at most $n$. We will further argue that there is at most one edge between any two connected components. Suppose for the sake of contradiction that there are two components of $(V, C)$ with two different edges connecting them, say $\{u, v\}$ and $\{u', v'\}$, where $u$ and $u'$ lie in the same connected component and $v$ and $v'$ in the other. As the diameter of each component is at most $k-1$, it follows that in $G$ there is a path from $u$ to $u'$ of length at most $k-1$, and a path from $v$ to $v'$ of length at most $k-1$. Together with the two edges connecting the components we obtain a cycle of length at most $2(k-1)+2=2k$, contradicting the assumption that $G$ has girth $2k+1$.

Therefore, the resulting graph after the contraction has $\Omega(m) = \Omega(n^{1+1/k})$ edges. ∎

**7.2. Additive contractions.** Turning to the case of a purely additive error, we obtain the following two results.

Theorem 14. *Let $G$ be a graph with unit length edges.*

(i) *For any even integer $0 \leq k \leq n$, the set of edges incident to the $k/2$ vertices of highest degrees is a $(1,k)$-contraction $C$ in $G$ with $\Phi(C) \geq km/(2n)$.*

(ii) *For any real number $0 < k \leq n$, the set of edges incident to two vertices of degree at least $n/k$ is a $(1,k)$-contraction $C$ in $G$ such that $G/C$ has $\mathcal{O}(n^2/k)$ edges.*

*These contractions can be computed in time $\mathcal{O}(m)$.*

As mentioned in the introduction, Bernstein and Chechik analyzed the contraction of Theorem 14(ii) in [9] and used it in their dynamic shortest paths algorithm, so this part is already proved.

*Proof of Theorem* 14(i). Let $U$ be the set of $k$ vertices in $G$ of highest degree. Then we have

$$\sum_{u\in U}\deg(u) \geq k/n \sum_{v\in V}\deg(v) = k/n \cdot 2m = 2km/n.$$

Let $C$ be the set of edges incident to any vertex in $U$. As each edge is incident to at most two vertices in $U$, we get $|C| \geq 1/2 \sum_{u\in U}\deg(u) \geq km/n$ from the previous inequality. As no shortest path visits a vertex in $U$ twice, $C$ is indeed a $(1,2k)$-contraction. The set $C$ can be computed as follows: We first compute the degrees of all vertices in time $\mathcal{O}(m)$, then find the $k$th largest element in this list in time $\mathcal{O}(n)$, and by another linear time sweep over this list we select $k$ vertices of highest degree. Overall, the required time is $\mathcal{O}(m)$. ∎

This result implies that the number of edges in $G/C$ is at most $m - km/n$. If $G$ is a path, no $(1,2k)$-contraction has an objective value greater than $2k$, and $km/n = k(1-1/n)$, showing that the objective value in Theorem 14(i) can be improved by at most a factor of two.

The information theoretic lower bound in [1] implies that for all $\varepsilon > 0$, any contraction $C$ such that $G/C$ has $\mathcal{O}(n^{4/3-\varepsilon})$ edges does not admit a constant additive error.

**7.3. Vertex reduction.** All of the results above show that contractions can be effectively used to reduce the number of edges in a dense graph. But one possible advantage of using a contraction instead of a spanner is that it also has the potential to reduce the number of vertices in the graph. Unfortunately, for constant approximation errors, it is not possible to guarantee more than a constant-factor reduction in general graphs: it is not hard to see that given a path on $n$ vertices, any $(k,1)$-contraction will still result in at least $n/(k+1)$ vertices. The same problem applies to general dense graphs, since they could still contain a long path within them. That being said, it seems likely that in practice contraction can lead to significant vertex reduction in many dense graphs. We ground this practical intuition with the following theoretical result for the special case of graphs with large minimum degree.

THEOREM 15. *Let $D$ be an integer. Any graph $G$ with minimum degree at least $D$ has a $(5,1)$-contraction $C$ such that the contracted graph $G/C$ has at most $n/D$ vertices, and such a contraction can be computed in time $\mathcal{O}(m)$.*

*Proof.* Recall the definition of an $r$-partition. For a cluster $P_i$ with center vertex $p_i$ we refer to $r$ as the *radius* of that cluster. This is the maximum distance of all cluster vertices from $p_i$.

We will show how to construct a 3-partition in which the number of clusters $P_i$ is at most $n/D$. Using the exact same argument as in the proof of Theorem 11, such a 3-partition yields the desired $(5,1)$-contraction. Our construction first builds clusters of radius 1, and then extends them to clusters of radius 2. The clustering with radius 1 proceeds very similarly to the proof of Lemma 7.1 before with $r = 1$. The crucial difference is that we choose as center vertices only vertices with degree at least $D$. If no such vertices are left, the clustering process terminates, and the remaining unclustered vertices have degree strictly less than $D$. It is easy to see that since those vertices have degree at least $D$ in the original graph, they must be adjacent to a vertex in a radius 1 cluster. We can thus assign each of those vertices to such a cluster arbitrarily, yielding a clustering of all vertices of $G$ with radius 2.

The number of clusters is at most $n/D$ because by construction every cluster contains at least $D$ vertices. This shows that the number of vertices in the contracted graph is at most $n/D$.

This algorithm can be implemented in time $\mathcal{O}(m)$ by using an adjacency list representation where we keep track of degree information after removing an edge from the graph. $\square$

To see that we cannot guarantee less than $n/D$ vertices, even with larger approximation error, consider the graph $G$ that consists of $n/D$ isolated $D$-cliques. We now show that even if $G$ is connected, we cannot guarantee $o(n/D)$ vertices in the contracted graph, even if we allow a larger (constant) approximation error.

THEOREM 16. *Let $D$ and $k$ be integers. There exists an infinite family of $n$-vertex graphs $G$ with minimum degree $D$ such that any $(k,1)$-contraction $C$ results in a graph $G/C$ with $n/((k+1)D)$ vertices.*

*Proof.* Assume for simplicity that $n$ is divisible by $D$. We construct the graph $G$ as follows. We partition the $n$ vertices into $n/D$ layers, with each layer containing exactly $D$ vertices. For $1 \leq i < n/D$, all vertices in layer $i$ receive an edge to all vertices in layer $i+1$. Clearly all vertices in the resulting graph have degree at least $D$. Let $u$ and $v$ be two vertices in layers $i$ and $j$, respectively. Then clearly we have $\mathrm{dist}_\ell(u,v) \geq |j-i|$. Now let $C$ be any $(k,1)$-contraction on $G$, and consider the connected components of the graph $(V,C)$. Applying (1) shows that $\mathrm{dist}_\ell(u,v) \leq k$ holds for any two vertices $u$ and $v$ in the same component. Combining these two inequalities shows that every connected component contains vertices from at most $k+1$ layers. As there are $n/D$ layers, the contracted graph has at least $n/((k+1)D)$ vertices. $\square$

**Acknowledgments.** We thank Martin Skutella for stimulating discussions about the problems treated in this paper. We also thank the anonymous referees for their valuable suggestions that helped improve the presentation of results.

## REFERENCES

[1] A. Abboud and G. Bodwin, *The 4/3 additive spanner exponent is tight*, in STOC'16—Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, ACM, New York, 2016, pp. 351–361.

[2] D. Aingworth, C. Chekuri, P. Indyk, and R. Motwani, *Fast estimation of diameter and shortest paths (without matrix multiplication)*, SIAM J. Comput., 28 (1999), pp. 1167–1181, https://doi.org/10.1137/S0097539796303421.

[3] I. Althöfer, G. Das, D. Dobkin, D. Joseph, and J. Soares, *On sparse spanners of weighted graphs*, Discrete Comput. Geom., 9 (1993), pp. 81–100, https://doi.org/10.1007/BF02189308.

[4] B. Awerbuch, *Complexity of network synchronization*, J. ACM, 32 (1985), pp. 804–823, https://doi.org/10.1145/4221.4227.

[5] D. A. Bader, H. Meyerhenke, P. Sanders, and D. Wagner, eds., *Graph Partitioning and Graph Clustering*, Contemp. Math. 588, AMS, Providence, RI, 2012.

[6] J. J. Bartholdi, III, J. B. Orlin, and H. D. Ratliff, *Cyclic scheduling via integer programs with circular ones*, Oper. Res., 28 (1980), pp. 1074–1085, https://doi.org/10.1287/opre.28.5.1074.

[7] S. Baswana, T. Kavitha, K. Mehlhorn, and S. Pettie, *New constructions of $(\alpha, \beta)$-spanners and purely additive spanners*, in Proceedings of the Sixteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SIAM, Philadelphia, 2005, pp. 672–681.

[8] S. Baswana and S. Sen, *A simple linear time algorithm for computing a $(2k-1)$-spanner of $O(n^{1+1/k})$ size in weighted graphs*, in Automata, Languages and Programming, Lecture Notes in Comput. Sci. 2719, Springer, Berlin, 2003, pp. 384–396, https://doi.org/10.1007/3-540-45061-0_32.

[9] A. Bernstein and S. Chechik, *Deterministic decremental single source shortest paths: Beyond the $O(mn)$ bound*, in STOC'16, Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, ACM, New York, 2016, pp. 389–397.

[10] A. Bernstein, K. Däubel, Y. Disser, M. Klimm, T. Mütze, and F. Smolny, *Distance-preserving graph contractions*, in 9th Innovations in Theoretical Computer Science Conference, ITCS 2018, 2018, Cambridge, MA, Schloss Dagstuhl–Leibniz-Zentrum für Informatik, Wadern, Germany, 2018, 51, https://doi.org/10.4230/LIPIcs.ITCS.2018.51.

[11] A. Bernstein, K. Däubel, Y. Disser, M. Klimm, T. Mütze, and F. Smolny, *Distance-preserving graph contractions*, preprint, arXiv:1705.04544, 2019.

[12] T. C. Biedl, B. Brejová, and T. Vinař, *Simplifying flow networks*, in Mathematical Foundations of Computer Science 2000 (Bratislava), Lecture Notes in Comput. Sci. 1893, Springer, Berlin, 2000, pp. 192–201, https://doi.org/10.1007/3-540-44612-5_15.

[13] G. Bodwin, *Linear size distance preservers*, in Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SIAM, Philadelphia, 2017, pp. 600–615, https://doi.org/10.1137/1.9781611974782.39.

[14] G. Bodwin and V. Vassilevska Williams, *Better distance preservers and additive spanners*, in Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SIAM, Philadelphia, 2016, pp. 855–872, https://doi.org/10.1137/1.9781611974331.ch61.

[15] T.-H. H. Chan, D. Xia, G. Konjevod, and A. Richa, *A tight lower bound for the Steiner point removal problem on trees*, in Approximation, Randomization and Combinatorial Optimization, Lecture Notes in Comput. Sci. 4110, Springer, Berlin, 2006, pp. 70–81, https://doi.org/10.1007/11830924_9.

[16] Y. K. Cheung, G. Goranci, and M. Henzinger, *Graph minors for preserving terminal distances approximately - lower and upper bounds*, in 43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016, 2016, Rome, Italy, Schloss Dagstuhl–Leibniz-Zentrum für Informatik, Wadern, Germany, 2016, 131, https://doi.org/10.4230/LIPIcs.ICALP.2016.131.

[17] E. Chlamtáč, M. Dinitz, G. Kortsarz, and B. Laekhanukit, *Approximating spanners and directed Steiner forest: Upper and lower bounds*, in Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SIAM, Philadelphia, 2017, pp. 534–553, https://doi.org/10.1137/1.9781611974782.34.

[18] T. Chu, Y. Gao, R. Peng, S. Sachdeva, S. Sawlani, and J. Wang, *Graph sparsification, spectral sketches, and faster resistance computation, via short cycle decompositions*, in 59th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2018, Paris, France, 2018, IEEE, Piscataway, NJ, 2018, pp. 361–372, https://doi.org/10.1109/FOCS.2018.00042.

[19] D. Coppersmith and M. Elkin, *Sparse sourcewise and pairwise distance preservers*, SIAM J. Discrete Math., 20 (2006), pp. 463–501, https://doi.org/10.1137/050630696.

[20] M. Dinitz, R. Krauthgamer, and T. Wagner, *Towards resistance sparsifiers*, in Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, vol. 40 of LIPIcs. Leibniz Int. Proc. Inform., Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Wadern, Germany, 2015, pp. 738–755.

[21] F. Dörfler and F. Bullo, *Kron reduction of graphs with applications to electrical networks*, IEEE Trans. Circuits Syst. I. Regul. Pap., 60 (2013), pp. 150–163, https://doi.org/10.1109/TCSI.2012.2215780.

[22] D. Durfee, R. Kyng, J. Peebles, A. B. Rao, and S. Sachdeva, *Sampling random spanning trees faster than matrix multiplication*, in STOC'17—Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, ACM, New York, 2017, pp. 730–742.

[23] P. Erdős, *Extremal problems in graph theory*, in Theory of Graphs and its Applications (Proc. Sympos. Smolenice, 1963), Czechoslovak Academy of Science, Prague, 1964, pp. 29–36.

[24] A. Gupta, *Steiner points in tree metrics don't (really) help*, in Proceedings of the Twelfth Annual ACM-SIAM Symposium on Discrete Algorithms (Washington, DC, 2001), SIAM, Philadelphia, 2001, pp. 220–227.

[25] C. Hübler, H.-P. Kriegel, K. M. Borgwardt, and Z. Ghahramani, *Metropolis algorithms for representative subgraph sampling*, in Proceedings of the 8th IEEE International Conference on Data Mining (ICDM 2008), 2008, Pisa, Italy, IEEE, Piscataway, NJ, 2008, pp. 283–292, https://doi.org/10.1109/ICDM.2008.124.

[26] R. M. Karp, *Reducibility among combinatorial problems*, in Complexity of Computer Computation, Plenum, New York, 1972, pp. 85–103.

[27] G. Kortsarz, *On the hardness of approximating spanners*, Algorithmica, 30 (2001), pp. 432–450, https://doi.org/10.1007/s00453-001-0021-y.

[28] G. Kortsarz and D. Peleg, *Generating sparse* 2*-spanners*, J. Algorithms, 17 (1994), pp. 222–236, https://doi.org/10.1006/jagm.1994.1032.

[29] R. Krauthgamer, H. L. Nguyễn, and T. Zondiner, *Preserving terminal distances using minors*, SIAM J. Discrete Math., 28 (2014), pp. 127–141, https://doi.org/10.1137/120888843.

[30] R. Kyng and S. Sachdeva, *Approximate Gaussian elimination for Laplacians—fast, sparse, and simple*, in 57th Annual IEEE Symposium on Foundations of Computer Science—FOCS 2016, IEEE Computer Society, Los Alamitos, CA, 2016, pp. 573–582.

[31] A. L. Liestman and T. C. Shermer, *Additive graph spanners*, Networks, 23 (1993), pp. 343–363, https://doi.org/10.1002/net.3230230417.

[32] E. Misiołek and D. Z. Chen, *Efficient algorithms for simplifying flow networks*, in Computing and Combinatorics, Lecture Notes in Comput. Sci. 3595, Springer, Berlin, 2005, pp. 737–746, https://doi.org/10.1007/11533719_75.

[33] D. Peleg and A. A. Schäffer, *Graph spanners*, J. Graph Theory, 13 (1989), pp. 99–116, https://doi.org/10.1002/jgt.3190130114.

[34] R. Wenger, *Extremal graphs with no $C^4$'s, $C^6$'s, or $C^{10}$'s*, J. Combin. Theory Ser. B, 52 (1991), pp. 113–116, https://doi.org/10.1016/0095-8956(91)90097-4.

[35] F. Zhou, S. Mahler, and H. Toivonen, *Network simplification with minimal loss of connectivity*, in ICDM 2010, 10th IEEE International Conference on Data Mining, Sydney, Australia, IEEE Computer Society, Los Alamitos, CA, 2010, pp. 659–668, https://doi.org/10.1109/ICDM.2010.133.

[36] D. Zuckerman, *Linear degree extractors and the inapproximability of max clique and chromatic number*, Theory Comput., 3 (2007), pp. 103–128, https://doi.org/10.4086/toc.2007.v003a006.