Almost Tight Lower Bounds for Hard Cutting Problems in Embedded Graphs

VINCENT COHEN-ADDAD, Sorbonne Universités, UPMC Univ Paris 06, CNRS, LIP6, Paris, France
ÉRIC COLIN DE VERDIÈRE, LIGM, CNRS, Univ Gustave Eiffel, Marne-la-Vallée, France
DÁNIEL MARX, CISPA Helmholtz Center for Information Security, Germany
ARNAUD DE MESMAY, LIGM, CNRS, Univ Gustave Eiffel, Marne-la-Vallée, France

We prove essentially tight lower bounds, conditionally to the Exponential Time Hypothesis, for two fundamental but seemingly very different cutting problems on surface-embedded graphs: the Shortest Cut Graph problem and the Multiway Cut problem.

A cut graph of a graph G embedded on a surface $\mathcal{S}$ is a subgraph of G whose removal from $\mathcal{S}$ leaves a disk. We consider the problem of deciding whether an unweighted graph embedded on a surface of genus g has a cut graph of length at most a given value. We prove a time lower bound for this problem of $n^{\Omega(g/\log g)}$ conditionally to the ETH. In other words, the first $n^{O(g)}$-time algorithm by Erickson and Har-Peled [SoCG 2002, Discr. Comput. Geom. 2004] is essentially optimal. We also prove that the problem is W[1]-hard when parameterized by the genus, answering a 17-year-old question of these authors.

A multiway cut of an undirected graph G with t distinguished vertices, called *terminals*, is a set of edges whose removal disconnects all pairs of terminals. We consider the problem of deciding whether an unweighted graph G has a multiway cut of weight at most a given value. We prove a time lower bound for this problem of $n^{\Omega(\sqrt{gt+g^2+t}/\log(g+t))}$, conditionally to the ETH, for any choice of the genus $g \geq 0$ of the graph and the number of terminals $t \geq 4$. In other words, the algorithm by the second author [Algorithmica 2017] (for the more general multicut problem) is essentially optimal; this extends the lower bound by the third author [ICALP 2012] (for the planar case).

Reductions to planar problems usually involve a gridlike structure. The main novel idea for our results is to understand what structures instead of grids are needed if we want to exploit optimally a certain value g of the genus.

A preliminary version of this paper appeared in *Proceedings of the 35th International Symposium on Computational Geometry*, 2019.

The first and the fourth authors are partially supported by the French ANR project ANR-18-CE40-0004-01 (FOCAL). The second and the fourth authors are partially supported by the French ANR project ANR-17-CE40-0033 (SoS) and the French ANR project ANR-19-CE40-0014 (MIN-MAX). The third author is supported by ERC Consolidator Grant SYSTEMATIC-GRAPH (No. 725978). The fourth author is partially supported by the French ANR project ANR-16-CE40-0009-01 (GATO) and the CNRS PEPS project COMP3D. Parts of this work were realized when he was working at GIPSA-lab in Grenoble.

Authors' addresses: V. Cohen-Addad, Brandschenkestrasse 110, 8002 Zürich, Switzerland; email: vcohenad@gmail.com; É. Colin de Verdière and A. de Mesmay, LIGM, Université Gustave Eiffel, 5 boulevard Descartes, Champs-sur-Marne, 77454 Marne-la-Vallée Cedex 2, France; emails: eric.colindeverdiere@u-pem.fr, ademesmay@gmail.com; D. Marx, CISPA Helmholtz Center for Information Security, Saarland Informatics Campus, Stuhlsatzenhaus 5, 66123 Saarbrücken, Germany; email: marx@cispa.de.

0004-5411/2021/07-ART30 $15.00
https://doi.org/10.1145/3450704

CCS Concepts: • **Mathematics of computing** → *Graphs and surfaces*; *Graph algorithms*; • **Theory of computation** → *Fixed parameter tractability*; *Computational geometry*;

Additional Key Words and Phrases: W[1]-hardness, Exponential Time Hypothesis, multiway cut, multicut, cut graph, surface

ACM Reference format:
Vincent Cohen-Addad, Éric Colin de Verdière, Dániel Marx, and Arnaud de Mesmay. 2021. Almost Tight Lower Bounds for Hard Cutting Problems in Embedded Graphs. *J. ACM* 68, 4, Article 30 (July 2021), 26 pages.
https://doi.org/10.1145/3450704

1 INTRODUCTION

Since the early 2010s, there has been a flurry of works investigating the complexity of solving exactly optimization problems on planar graphs, leading to what was coined as the "square root phenomenon" by Marx [31]: Many problems turn out to be easier on planar graphs, and the improvement compared to the general case is captured exactly by a square root. For instance, problems solvable in time $2^{O(n)}$ in general graphs can be solved in time $2^{O(\sqrt{n})}$ in planar graphs, and similarly, in a parameterized setting, FPT problems admitting $2^{O(k)}n^{O(1)}$-time algorithms or W[1]-hard problems admitting $n^{O(k)}$-time algorithms can often be sped up to $2^{\widetilde{O}(\sqrt{k})}n^{O(1)}$ and $n^{\widetilde{O}(\sqrt{k})}$, respectively, when restricted to planar graphs. We have many examples where matching upper bounds (algorithms) and lower bounds (complexity reductions) show that indeed the best possible running time for the problems has this form. On the side of upper bounds, the improvement often stems from the fact that planar graphs have planar separators (and thus treewidth) of size $O(\sqrt{n})$, and the theory of bidimensionality provides an elegant framework for a similar speedup in the parameterized setting for some problems [11]. However, in many cases these algorithms rely on highly problem-specific arguments [2, 5, 14, 24, 26, 33, 34]. The lower bounds are conditional to the **Exponential Time Hypothesis (ETH)** of Impagliazzo, Paturi, and Zane [21] and follow from careful reductions from problems displaying this phenomenon, e.g., Planar 3-Coloring, k-Clique, or Grid Tiling. We refer to the recent book [8] for precise results along these lines.

While the theme of generalizing algorithms from planar graphs to surface-embedded graphs has attracted a lot of attention, and has flourished into an established field mixing algorithmic and topological techniques (see Reference [6]), the same cannot be said at all of the lower bounds. Actually, up to our knowledge, there are very few works explicitly establishing algorithmic lower bounds based on the genus of the surfaces on which a graph is embedded, or even just hardness results when parameterized by the genus—the only ones we are aware of are the exhaustive treatise [32] of Marx and Pilipczuk on Subgraph Isomorphism, where some of the hardness results feature the genus of the graph, the lower bounds of Curticapean and Marx [7] on the problem of counting perfect matchings, and the work of Chen, Kanj, Perković, Sedgwick, and Xia [1].

In this work, we address this surprising gap by providing lower bounds conditioned on the ETH for two fundamental yet seemingly very different cutting problems on surface-embedded graphs: the Shortest Cut Graph problem and the Multiway Cut problem. In both cases, our lower bounds match the best known algorithms up to a logarithmic factor in the exponent. We believe that the tools that we develop in this article could pave the way towards establishing lower bounds for other problems on surface-embedded graphs.

1.1 The Shortest Cut Graph Problem

A *cut graph* of an edge-weighted graph G cellularly embedded on a surface $\mathcal{S}$ is a subgraph of G that has a unique face, which is a disk (note that such a cut graph only exists when G is cellularly embedded). Computing a shortest cut graph is a fundamental problem in algorithm design, as it is often easier to work with a planar graph than with a graph embedded on a surface of positive genus, since the large toolbox that has been designed for planar graphs becomes available. Furthermore, making a graph planar is useful for various purposes in computer graphics and mesh processing, see, e.g., Reference [38]. Be it for a practical or a theoretical goal, a natural measure of the distortion induced by the cutting step is the length of the topological decomposition.

Thus, the past decade has witnessed a lot of effort on how to obtain efficient algorithms for the problems of computing short topological decompositions, see, for example, the survey in Reference [6]. For the shortest cut graph problem, Erickson and Har-Peled [13] showed that the problem is NP-hard when the genus is considered part of the input and gave an exact algorithm running in time $n^{O(g)}$, where n is the size of the input graph and g the genus of the surface, together with an $O(\log^2 g)$-approximation running in time $O(g^2 n \log n)$. Cohen-Addad and de Mesmay [4] gave a $(1+\varepsilon)$-approximation algorithm running in time $f(\varepsilon, g)n^3$, where f is some explicit computable function. Whether it is possible to improve upon the exact algorithm of Erickson and Har-Peled by designing an FPT algorithm for the problem, namely an exact algorithm running in time $f(g)n^{O(1)}$, has been raised by these authors [13, Conclusion] and has remained an open question over the last 17 years.

In this article, we solve this question by proving that the result of Erickson and Har-Peled cannot be significantly improved. We indeed show a lower bound of $n^{\Omega(g/\log g)}$ (for the associated decision problem, even in the unweighted case) assuming the ETH of Impagliazzo, Paturi, and Zane [21] (see Definition 2.4), and also prove that the problem is W[1]-hard. More formally, we consider the following decision problem:

Shortest Cut Graph:
Input: An unweighted graph G with n vertices and edges cellularly embedded on an orientable surface of genus g, and an integer λ.
Output: Whether G admits a cut graph of length at most λ.

Theorem 1.1.

1. *The Shortest Cut Graph problem is W[1]-hard when parameterized by g.*
2. *Assuming the ETH, there exists a universal constant $\alpha_{\mathrm{CG}} > 0$ such that for any fixed integer $g \geq 0$, there is no algorithm solving all the Shortest Cut Graph instances of genus at most g in time $O(n^{\alpha_{\mathrm{CG}} \cdot (g+1)/\log(g+2)})$.*

In the second item, the strange-looking additive constants are just here to ensure that the theorem is still correct for the values $g = 0$ and $g = 1$.

Remark 1.2. Let us observe that it is sufficient to prove Theorem 1.1 in the case where the genus g is assumed to be at least some constant $\bar{g}$. Indeed, assume we did so. Shortest Cut Graph admits a trivial, unconditional, lower bound of $\Omega(n)$, as any algorithm has to read the input. Therefore, if we take α_{CG} small enough so that $\alpha_{\mathrm{CG}} \cdot (g+1)/\log(g+2) < 1$ for any $g \leq \bar{g}$, then this trivial lower bound finishes the proof for the remaining constant number of values. The exact same remark will apply to the proofs of most theorems establishing lower bounds in the article: It will suffice to prove them for the case when the parameter is larger than a certain constant, the remaining cases being trivial.

1.2 The Multiway Cut Problem

The second result of our article concerns the Multiway Cut problem (also known as the Multiterminal Cut problem). Given an edge-weighted graph G together with a subset T of t vertices called terminals, a multiway cut is a set of edges whose removal disconnects all pairs of terminals. Computing a minimum-weight multiway cut is a classic problem that generalizes the minimum $s - t$ cut problem and some closely related variants have been actively studied since as early as 1969 [20]. On general graphs, while the problem is polynomial-time solvable for $t = 2$, it becomes NP-hard for any fixed $t \geq 3$, see Reference [9]. In the case of planar graphs, it remains NP-hard if t is arbitrarily large but can be solved in time $2^{O(t)} n^{O(\sqrt{t})}$, where n is the number of vertices and edges of the graph [24], and a lower bound of $n^{\Omega(\sqrt{t})}$ was proved (conditionally on the ETH) by Marx [30]. A generalization to higher-genus graphs was recently obtained by Colin de Verdière [5] who devised an algorithm running in time[1] $f(g, t) \cdot n^{O(\sqrt{gt+g^2+t})}$ in graphs of genus g, for some function f. If one allows some approximation, then this can be significantly improved: Cohen-Addad, Colin de Verdière, and de Mesmay recently provided a $(1 + \varepsilon)$-approximation algorithm running in time $f(\varepsilon, g, t) \cdot n \log n$ [3]. The latter two results are actually valid for the more general Multicut problem, in which one looks for a minimum-weight set of edges whose removal disconnects some specified pairs of terminals (but not necessarily all of them, as opposed to Multiway cut).

We prove a lower bound of $n^{\Omega(\sqrt{gt+g^2+t}/\log(g+t))}$ for the associated decision problem, even in the unweighted case, which almost matches the aforementioned best known upper bound, and generalizes the lower bound of Marx [30] for the planar case. Actually, we prove a lower bound that holds for any value of the integers g and t as long as $t \geq 4$. More formally, we consider the following decision problem:

Multiway Cut:
Input: An unweighted graph G with n vertices and edges, a set T of vertices, and an integer λ.
Output: Whether there exists a multiway cut of (G, T) of value at most λ.

Theorem 1.3. *Assuming the ETH, there exists a universal constant $\alpha_{\mathrm{MC}} > 0$ such that for any fixed choice of integers $g \geq 0$ and $t \geq 4$, there is no algorithm that decides all the Multiway Cut instances (G, T, λ) for which G is embeddable on the orientable surface of genus g and $|T| \leq t$, in time $O(n^{\alpha_{\mathrm{MC}} \sqrt{gt+g^2+t}/\log(g+t)})$.*

Since Multicut is a generalization of Multiway Cut, the lower bound also holds for Multicut. Note that taking $g = 0$ in this theorem yields lower bounds for the Planar Multiway Cut problem, and recovers, up to a logarithmic factor in the exponent, the lower bounds obtained by Marx [30] for that problem. In the opposite regime, we also prove W[1]-hardness with respect to the genus for instances with four terminals, see Proposition 4.1. We remark that $t = 2$ corresponds to the minimum cut problem, which is polynomial-time solvable, so a lower bound on t is necessary. While the last remaining case, for $t = 3$, is known to be NP-hard [9], our techniques do not seem to encompass it, and we leave its parameterized complexity with respect to the genus as an open problem.

Parameterized lower bounds in the literature often have the form "assuming the ETH, for any function f, there is no $f(k)n^{o(h(k))}$ algorithm to solve problem X," where h is some specific

[1] Note that Reference [5] states the exponent slightly imprecisely in the form $O(\sqrt{gt + g^2})$, which is not correct for, e.g., $g = 0$. The current form (or, more precisely, an upper bound of $f(g, t) \cdot n^{c\sqrt{gt+g^2+t}}$ for some $c > 0$) is correct for every $g \geq 0$ and $t \geq 1$.

dependency on the parameter. The lower bounds that we prove in Theorems 1.1 and 1.3 are instead of the form "assuming the ETH, there exists a universal constant α such that for any fixed k, there is no $O(n^{\alpha h(k)})$ algorithm to solve problem X." The latter lower bounds imply the former: Indeed, $f(k)n^{o(h(k))} = O(n^{\alpha h(k)})$ for a fixed k. Our results are stronger, concerning instances for any fixed k. Moreover, lower bounds with two parameters are difficult to state with $o()$ notation. The statement of Theorem 1.3 handles every combination of the two parameters in a completely formal way.

While Theorem 1.3 does not use an embedded graph as an input, we can find an embedding of a graph on a surface with minimum possible genus in $2^{\text{poly}(g)}n$ time [23, 36]. Thus, the same hardness result holds in the embedded case and the question is not about whether we are given the embedding or not.

1.3 Main Ideas of the Proof

What is a good starting problem to prove hardness results for surface-embedded graphs? For planar graphs, the Grid Tiling problem of Marx [28] has now emerged as a convenient, almost universal, tool to establish parameterized hardness results and precise lower bounds based on the ETH. A similar approach, based on **constraint satisfaction problems (CSPs)** on d-dimensional grids, was used by Marx and Sidiropoulos [35] to obtain lower bounds for geometric problems on low-dimensional Euclidean inputs (see also Reference [10] for a similar framework for geometric intersection graphs). However, these techniques do not apply directly for the problems that we consider. Indeed, the bounds implied by these approaches are governed by the treewidths of the underlying graphs and are of the type $n^{\Omega(\sqrt{p})}$ or $n^{\Omega(p^{1-1/d})}$, respectively, where p is the parameter of interest and d the dimension of the grid in the latter case. In contrast, here, we are looking for bounds of the form $n^{\Omega(p/\log p)}$ (while this is not apparent from looking at Theorem 1.3, this also turns out to be the main regime of interest for the Multiway Cut problem).

Our first contribution, in Section 3, is to introduce a new hard problem for embedded graphs, which is versatile enough to be used as a starting point to obtain lower bounds for both the Shortest Cut Graph and the Multiway Cut problem (and hopefully others). It is a variant of the Grid Tiling problem that we call 4-Regular Graph Tiling; in a precise sense, it generalizes the Grid Tiling problem to allow for embedded 4-regular graphs different from the planar grid to be used as the structure graph of the problem. We show that a CSP instance with k binary constraints can be simulated by a 4-Regular Graph Tiling instance with parameter k. A result of Marx [29] shows that, assuming the ETH, such CSP instances cannot be solved in time $f(k)n^{\Omega(k/\log k)}$, giving a similar lower bound for 4-Regular Graph Tiling (Theorem 3.1).

We then establish in Sections 4 and 5 the lower bounds for the Shortest Cut Graph and "one half" of the lower bound for Multiway Cut, namely, for the regime where the genus dominates the number of terminals, in which case we prove a lower bound of $n^{\Omega(g/\log g)}$. Both reductions proceed from 4-Regular Graph Tiling and use as a building block an intricate set of *cross gadgets* originally designed by Marx [30] for his hardness proof of the Planar Multiway Cut problem. While it does not come as a surprise that these gadgets are useful for Multiway Cut instances in the case of surface-embedded graphs, for which planar tools can often be used, it turns out that via basic planar duality, they also provide exactly the needed technical tool for establishing the hardness of Shortest Cut Graph.

The "second half" of the lower bound in Theorem 1.3 is in the regime where the number of terminals dominates the genus, for which we establish a lower bound of $n^{\Omega(\sqrt{gt+t}/\log(g+t))}$. In Section 6, we use a similar strategy as before but bypass the use of the 4-Regular Graph Tiling problem. Instead, we rely directly on the aforementioned theorem of Marx on the parameterized hardness of CSPs, which we apply not to a family of expanders, but to blow-ups of expanders, i.e., expanders where vertices are replaced by grids of a well-chosen size. This size is prescribed exactly by the tradeoff between the genus and the number of terminals, as described with the two integers g and t in Theorem 1.3. The key property of these blow-ups is that their treewidth is $\mathrm{tw} = \Theta(\sqrt{gt + t})$ and thus the $n^{\Omega(\mathrm{tw}/\log \mathrm{tw})}$ lower bound on the complexity of CSPs with these blow-ups as primal graphs yields exactly the target lower bound. The reduction from CSPs to Multiway Cut is carried out in Proposition 6.1 and also relies on cross gadgets.

The proof of Theorem 1.3, in Section 7, results from the two halves given by Propositions 4.1 and 6.1.

Sections 4, 5, and 6 are roughly put in increasing order of technical difficulty, but they are independent; the reader interested in Theorem 1.3 can safely skip Section 5, while the reader interested in Theorem 1.1 can safely skip Sections 4, 6, and 7.

2 PRELIMINARIES

2.1 Graphs and Surfaces

For the sake of convenience in the proofs, unless otherwise noted, the graphs in this article are not necessarily simple; they may have loops and multiple edges. However, our hardness results also hold for simple graphs, because the instances of Shortest Cut Graph or Multiway Cut can easily be made simple by subdividing edges if desired.

For extensive background on graphs on surfaces, we refer to the classic textbook of Mohar and Thomassen [37]. Throughout this article, we only consider ***surfaces*** that are compact, connected, and orientable. By the classification theorem of surfaces, each such surface $\mathcal{S}$ is homeomorphic to a sphere with g handles attached and b disks removed; g is called the ***genus*** of the surface and b its number of ***boundaries***. A ***path***, or ***curve***, is a continuous map from $[0, 1]$ to $\mathcal{S}$. A path is ***simple*** if it is injective.

An ***embedding*** of G on $\mathcal{S}$ is a crossing-free drawing of G on $\mathcal{S}$, i.e., the images of the vertices are pairwise distinct and the image of each edge is a simple path intersecting the image of no other vertex or edge, except possibly at its endpoints. When embedding a graph on a surface with boundaries, we adopt the convention that while vertices can be mapped to a boundary, interiors of edges cannot. A ***face*** of the embedding is a connected component of the complement of the graph. A ***cellular embedding*** is an embedding of a graph where every face is a topological disk. By a slight abuse of language, we will often identify an abstract graph with its embedding. If G is a graph embedded on $\mathcal{S}$, then the surface obtained by ***cutting*** $\mathcal{S}$ along G is the disjoint union of the faces of G; it is an (*a priori* disconnected) surface with boundary.

Every graph embeddable on a surface of genus g is also embeddable on a surface of genus g', for all $g' \geq g$. A graph embedded on a surface of minimum possible genus is cellularly embedded. The genus of that surface is called the ***genus*** of the graph; it is at most the number of edges of the graph.

To a graph cellularly embedded on a surface without boundary $\mathcal{S}$, one can naturally associate a dual graph G^* embedded on $\mathcal{S}$, whose vertices are the faces of G and two such vertices are connected by an edge e^* for every edge e their dual faces share; e^* crosses e and no other edge of G.

2.2 Expanders, Separators, and Treewidth

For a *simple* graph G, we denote by $\boldsymbol{\lambda(G)}$ the second largest eigenvalue of its adjacency graph. If G is d-regular, then it is a basic fact that $\lambda(G) \leq d$. A family of d-regular ***expanders*** is an infinite family of d-regular simple graphs G such that $\lambda(G)/d < c_{\mathrm{exp}} < 1$ for some constant c_{exp}. A family $\mathcal{G}$ of graphs is ***dense*** if for any $n > 0$ there exists a graph in $\mathcal{G}$ with $\Theta(n)$ vertices (where the $\Theta()$ hides a universal constant).

Lemma 2.1. *There exists a dense family $\mathcal{H}$ of bipartite four-regular expanders.*

This lemma can be proved using a well-known simple probabilistic argument showing that random bipartite regular graphs are expanders, or with more intricate explicit constructions. We refer to the survey of Hoory, Linial, and Wigderson [19] or the groundbreaking recent works of Marcus, Spielman, and Srivastava [27].

The ***treewidth*** $\mathrm{tw}(G)$ of a graph G is a parameter measuring intuitively how close it is to a tree. Since we will use this parameter in a black-box manner and not rely precisely on its definition, we do not include it here and refer to graph theory textbooks, e.g., Diestel [12, Chapter 12]. We only indicate a few basic properties that we will use. For every graph G, we have $\mathrm{tw}(G) \leq |V(G)| - 1$. Every graph containing a $\delta \times \delta$-grid as a subgraph has treewidth at least δ.

For $\alpha < 1$, an ***α-separator*** of a graph G is a subset C of vertices of G such that each connected component of $G - C$ has a fraction at most α of the vertices of H.

Lemma 2.2 [8, Lemma 7.19]. *Let G be a graph with treewidth at most k. Then G has a $1/2$-separator of size at most $k + 1$.*

Lemma 2.3. *Let G be a simple d-regular graph. Then every $1/2$-separator of G has size at least $|V(G)|(d - \lambda(G))/16d$. Moreover, the treewidth of G is at least $\lfloor |V(G)| \cdot (d - \lambda(G))/8d \rfloor$.*

Proof. First, we remind some standard definitions. The edge expansion $h(G)$ of G and the (outer) vertex expansion $h_{\mathrm{out}}(G)$ of G are defined as follows, letting n be the number of vertices of G:

$$h(G) := \min_{\substack{A \subseteq V(G) \\ 1 \leq |A| \leq n/2}} \frac{|\{uv \in E(G) \mid u \in A, v \in \bar{A}\}|}{|A|}, \text{ and}$$

$$h_{\mathrm{out}}(G) := \min_{\substack{A \subseteq V(G) \\ 1 \leq |A| \leq n/2}} \frac{|\{v \in \bar{A} \mid \exists u \in A, uv \in E(G)\}|}{|A|}.$$

First, the "easy direction" of the Cheeger inequality gives $(d - \lambda(G))/2 \leq h(G)$, see for example Hoory, Linial, and Wigderson [19, Theorem 2.4]. Also, it is easy to see that $h(G) \leq d h_{\mathrm{out}}(G)$. So $(d - \lambda(G))/2d \leq h_{\mathrm{out}}(G)$.

Let us prove the bound on the size of $1/2$-separators. Let S be a $1/2$-separator for G. Assume that $|S| < n(d - \lambda(G))/16d$; in particular, $|S| \leq n/8$. Since S is a $1/2$-separator, there is a set A disjoint from S, of size between $n/4$ and $3n/4$, whose vertices are adjacent only to vertices in $S \cup A$. By replacing A with $V(G) - S - A$ if necessary, we can assume that A actually has size between $n/4 - |S| \geq n/8$ and $n/2$. Thus, $h_{\mathrm{out}}(G) \leq |S|/|A| \leq 8|S|/n$, which together with the bound of the previous paragraph implies the result.

To prove the lower bound on the treewidth, note that additionally to the above bound on h_{out}, we have $\lfloor n/4 \cdot h_{\mathrm{out}}(G) \rfloor \leq \mathrm{tw}(G)$, by Grohe and Marx [17, Proposition 1] (in which we choose $\alpha = 1/2$). □

2.3 Exponential Time Hypothesis

Our lower bounds are conditioned on the ETH, which was conjectured in Reference [21] and is stated below. ***3SAT*** denotes the Boolean satisfiability problem in which instances are presented in conjunctive normal form with at most three literals per clause.

Conjecture 2.4 (Exponential Time Hypothesis [21]). *There exists a positive real value $s > 0$ such that 3SAT, parameterized by n, has no $2^{sn}(n+m)^{O(1)}$-time algorithm, where n denotes the number of variables and m denotes the number of clauses.*

We refer to the survey [25] for background and discussion of this conjecture. Informally speaking, Conjecture 2.4 states that there is no algorithm for 3SAT that is subexponential in the number of *variables* of the formula. The Sparsification Lemma of Impagliazzo, Paturi, and Zane [22, Corollary 1] shows that the ETH is equivalent to saying that there is no algorithm subexponential in the *length* of the formula.

Corollary 2.5. *If the ETH holds, then there exists a positive real value $s > 0$ such that 3SAT has no $2^{s(n+m)}(n+m)^{O(1)}$-time algorithm, where n denotes the number of variables and m denotes the number of clauses.*

2.4 Constraint Satisfaction Problems

A ***binary constraint satisfaction problem*** is a triple (V, D, C) where

- V is a set of ***variables***,
- D is a ***domain*** of values,
- C is a set of ***constraints***, each of which is a triple of the form $\langle u, v, R\rangle$, where (u, v) is a pair of variables called the ***scope***, and R is a subset of D^2 called the ***relation***.

All the CSPs in this article will be binary, and thus we will omit the adjective binary. A solution to a constraint satisfaction problem instance is a function $f : V \to D$ such that for each constraint $\langle u, v, R\rangle$, the pair $(f(u), f(v))$ is a member of R. An algorithm ***decides*** a CSP instance I if it outputs true if and only if that instance admits a solution.

The ***primal graph*** of a CSP instance $I = (V, D, C)$ is a graph with vertex set V such that distinct vertices $u, v \in V$ are adjacent if and only if there is a constraint whose scope contains both u and v.

The starting points for the reductions in this article are the following two theorems, which state in a precise sense that the treewidth of the primal graph of a binary CSP establishes a lower bound on the best algorithm to decide it.

Theorem 2.6 ([16, 18]). *Let $\mathcal{G}$ be an arbitrary class of graphs with unbounded treewidth. Let us consider the problem of deciding the binary CSP instances whose primal graph, G, lies in $\mathcal{G}$. This problem is W[1]-hard parameterized by the treewidth of the primal graph.*

Theorem 2.7 ([29]). *Assuming the ETH, there exists a universal constant α_{CSP} such that for any fixed primal graph G such that $tw(G) \geq 2$, there is no algorithm deciding the binary CSP instances whose primal graph is G in time $O(|D|^{\alpha_{\text{CSP}} \cdot tw(G)/\log tw(G)})$.*

Theorem 2.6 is due to Grohe, Schwentick, and Segoufin [18] (see also Grohe [16]). Theorem 2.7 follows from the work of Marx [29]. Since this statement is stronger than the main theorem of [29] (which assumes the existence of an algorithm solving binary CSP instances for a class of graphs with unbounded treewidth), we explain how to prove it in the rest of this subsection. This reformulation could be useful also for other problems where lower bounds of this form are proved; it seems to be especially important for the clean handling of lower bounds with respect to two parameters in the exponent.

We first need to recall some definitions and results from Reference [29]. A graph H is a ***minor*** of G if H can be obtained from G by a sequence of vertex deletions, edge deletions, and edge contractions. Equivalently, a graph H is a minor of G if there is a *minor mapping* from H to G, which is a function $\psi : V(H) \to 2^{V(G)}$ satisfying the following properties: (1) $\psi(v)$ is a connected vertex set in G for every $v \in V(H)$, (2) $\psi(u) \cap \psi(v) = \emptyset$ for every $u \neq v$, and (3) if $uv \in E(H)$, then there is an edge of G intersecting both $\psi(u)$ and $\psi(v)$. Given a graph G and an integer q, we denote by $G^{(q)}$ the graph obtained by replacing every vertex with a clique of size q and replacing every edge with a complete bipartite graph on $q + q$ vertices. The main combinatorial result of Reference [29] is the following embedding theorem:

Theorem 2.8 ([29, Theorem 3.1]). *There are computable functions $f_1(G)$, $f_2(G)$, and a universal constant c such that for every $k \geq 2$, if G is a graph with $tw(G) \geq k$ and H is a graph with $|E(H)| = m \geq f_1(G)$ and no isolated vertices, then H is a minor of $G^{(q)}$ for $q = \lceil cm \log k/k \rceil$. Furthermore, such a minor mapping can be found in time $f_2(G)m^{O(1)}$.*

We will also need the following (fairly straightforward) reductions from Reference [29]:

Lemma 2.9. *Given an instance of 3SAT with n variables and m clauses, it is possible to construct in polynomial time an equivalent CSP instance with $n + m$ variables, $3m$ binary constraints, and domain size 3.*

Lemma 2.10. *Assume that G_1 is a minor of G_2. Given a binary CSP instance I_1 with primal graph G_1 and a minor mapping from G_1 to G_2, it is possible to construct in polynomial time an equivalent instance I_2 with primal graph G_2 and the same domain.*

Lemma 2.11. *Given a binary CSP instance $I_1 = (V_1, D_1, C_1)$ with primal graph $G^{(q)}$ (where G has no isolated vertices), it is possible to construct (in time polynomial in the size of the* output*) an equivalent instance $I_2 = (V_2, D_2, C_2)$ with primal graph G and $|D_2| = |D_1|^q$.*

With these tools at hand, we are ready to prove Theorem 2.7.

Proof of Theorem 2.7. Assume that the ETH holds. By Corollary 2.5, let $s > 0$ be a universal constant such that 3SAT has no $2^{s(n+m)}(n + m)^{O(1)}$-time algorithm, where n denotes the number of variables and m denotes the number of clauses.

Let $c \geq 1$ be a universal constant satisfying Theorem 2.8 and let us define $\alpha_{\mathrm{CSP}} = s/(6c \log_2 3)$. Let G be a graph with treewidth $k \geq 2$. Throughout this proof, we consider that G (and thus k) are fixed, in the sense that the $O()$ notation can hide factors depending on G and k. In contrast, the notation "poly" denotes a fixed polynomial, independent of G and k. Suppose that an algorithm $\mathbb{A}_G$ decides the binary CSP instances whose primal graph is G in time $O(|D|^{\alpha_{\mathrm{CSP}} \cdot k/\log k})$.

Consider an instance Ψ of 3SAT with n variables and m clauses. Using Lemma 2.9, we construct in $\mathrm{poly}(n + m)$ time an instance I_1 of CSP equivalent to Ψ with at most $n + m$ variables, at most $3m$ binary constraints, and domain size 3. Let H be the primal graph of I_1; without loss of generality, it does not have any isolated vertex. If $3m < \max\{f_1(G), k\}$, then we can solve I_1, and thus Ψ, in $O(1)$ time (hiding a factor depending on G and k). Otherwise, by Theorem 2.8, graph H is a minor of $G^{(q)}$ for $q = \lceil c(3m) \log k/k \rceil \leq 2c(3m) \log k/k$ (the inequality uses the facts that $c \geq 1$ and $3m \geq k \geq 2$), and the minor mapping can be found in time $f_2(G)\mathrm{poly}(m)$. Then, by Lemma 2.10, I_1 can be turned in time $\mathrm{poly}(n + m)$ into an instance I_2 with primal graph $G^{(q)}$ and domain size 3, which, by Lemma 2.11, can be turned in time $O(\mathrm{poly}(3^{2q}))$ into an instance I_3 with primal graph G and domain size 3^q. Using these reductions and algorithm $\mathbb{A}_G$ to solve I_3, we can solve Ψ in time $\mathrm{poly}(n + m) + 2^{\beta q} + O((3^q)^{\alpha_{\mathrm{CSP}} \cdot k/\log k})$, for some universal constant $\beta > 0$ that does not depend on k. By definition of q and α_{CSP}, this is

$$\mathrm{poly}(n + m) + 2^{\gamma m \log k/k} + O(2^{sm})$$

for some universal constant $\gamma > 0$ that does not depend on k.

Let $\bar{k} \geq 2$ be a universal constant such that $\gamma \log k/k \leq s$ for every $k \geq \bar{k}$; then the total running time becomes $O(2^{sm}\mathrm{poly}(n + m))$. Therefore, we have proved that if there is a graph G with $\mathrm{tw}(G) \geq \bar{k}$ and an algorithm $\mathbb{A}_G$ deciding the binary CSP instances with primal graph G in time $O(|D|^{\alpha_{\mathrm{CSP}} \cdot k/\log k})$, then we can solve any 3SAT instance in time $2^{s(n+m)}\mathrm{poly}(n + m)$, which, by our choice of s, implies that the ETH does not hold. By Remark 1.2, this suffices to conclude the proof. □

We will rely in particular on the following corollary of Theorems 2.6 and 2.7:

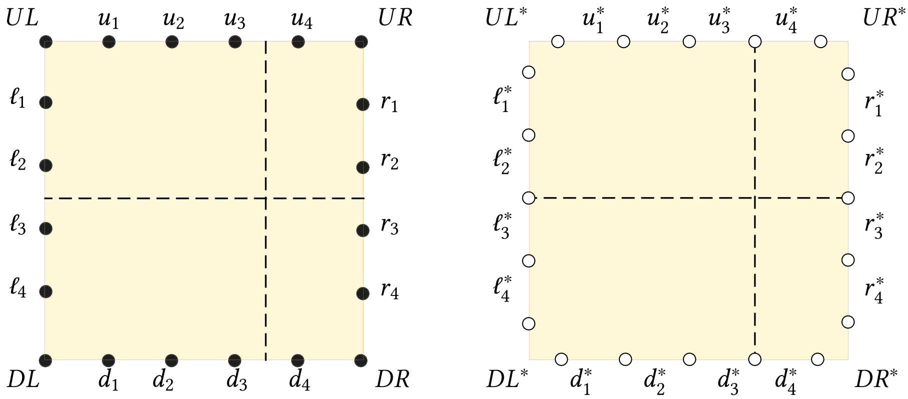

Fig. 1. Left: a cross gadget G_S for $\Delta = 3$. The dashed line indicates a multiway cut that represents the pair $(2, 3)$. Right: a dual cross gadget G_S^* for $\Delta = 3$. The dashed lined is a dual multiway cut that represents the pair $(2, 3)$.

Corollary 2.12.

(1) Deciding the binary CSP instances whose primal graph has at most k vertices, is four-regular and bipartite is W[1]-hard when parameterized by k.

(2) Assuming the ETH, there exists a universal constant α_{CSP} such that for any fixed $k \geq 2$, there is no algorithm deciding the binary CSP instances whose primal graph has at most k vertices, is four-regular and bipartite in time $O(|D|^{\alpha_{\text{REGCSP}} \cdot k/\log k})$.

Proof. For the first item, we apply Theorem 2.6 to the infinite family $\mathcal{H}$ of four-regular bipartite expanders output by Lemma 2.1. Then, Lemma 2.3 implies that for each such graph, the treewidth is linear in the number of vertices, and so we conclude that the problem is W[1]-hard when parameterized by k.

For the second item, by Lemma 2.1, there are universal constants $c > 0$, $\bar{k} > 0$, and $c_{\text{exp}} < 1$ such that, for any $k \geq \bar{k}$, there exists a four-regular bipartite expander P such that $\lambda(P)/4 < c_{\text{exp}}$, and with at least ck but at most k vertices. By Lemma 2.3, for some universal constant $c' > 0$, the treewidth of P is at least $c'k$, which we can assume to be at least two (up to increasing $\bar{k}$).

We choose α_{REGCSP} so that $\alpha_{\text{REGCSP}} \leq \alpha_{\text{CSP}}/c'$. Let us assume that there exists an algorithm as described by the corollary. Assume first that $k \geq \bar{k}$, and let P be obtained as in the previous paragraph. The assumed algorithm would in particular decide the binary CSP instances whose primal graph is P in time $O(|D|^{\alpha_{\text{REGCSP}} \cdot k/\log k}) = O(|D|^{\alpha_{\text{REGCSP}}/c' \cdot \text{tw}(P)/\log \text{tw}(P)}) = O(|D|^{\alpha_{\text{CSP}} \cdot \text{tw}(P)/\log \text{tw}(P)})$ (for fixed P, with $\text{tw}(P) \geq 2$). By Theorem 2.7, this would contradict the ETH.

Thus the second part of the corollary is proved for $k \geq \bar{k}$. The other cases are trivial if α_{REGCSP} is small enough, as in Remark 1.2 (because if $k \geq 2$, then there indeed exist some binary CSP instances whose primal graph has the form indicated in the statement of the corollary). □

2.5 Cross Gadgets

We rely extensively on the following intricate family of gadgets introduced by Marx in his proof of hardness of Planar Multiway Cut [30], which we call ***cross gadgets***; see Figure 1, left. Let Δ be an integer. The gadgets always have the form of a planar graph G_S embedded on a disk, with $4\Delta + 8$ distinguished vertices on its boundary, which are, in clockwise order, denoted by

$$UL, u_1, \ldots, u_{\Delta+1}, UR, r_1, \ldots, r_{\Delta+1}, DR, d_{\Delta+1}, \ldots d_1, DL, \ell_{\Delta+1}, \ldots, \ell_1.$$

The embedding is chosen so that the boundary of the disk intersects the graph precisely in this set of distinguished vertices; the interior of the edges lie in the interior of the disk. We consider the vertices UL, UR, DR, and DL as terminals in that gadget, and thus a multiway cut M of the gadget is a subset of the edges of G_S such that $G_S \setminus M$ has at least four components, and each of the terminals is in a distinct component. We say that a multiway cut M of the gadget ***represents*** the pair $(i, j) \in [\Delta]^2$ (where, as usual, $[\Delta]$ denotes the set $\{1, \dots, \Delta\}$) if $G_S \setminus M$ has exactly four components that partition the distinguished vertices into the following classes:

$$\{UL, u_1, \dots, u_j, \ell_1, \dots, \ell_i\} \qquad \{UR, u_{j+1}, \dots_{\Delta+1}, r_1, \dots, r_i\}$$
$$\{DL, d_1, \dots d_j, \ell_{i+1}, \dots, \ell_{\Delta+1}\} \qquad \{DR, d_{j+1}, \dots, d_{\Delta+1}, r_{i+1}, \dots r_{\Delta+1}\}$$

We remark that, as in the original article [30], the notation (i, j) is in matrix form (i.e., (row, column), with $(1, 1)$ being in the corner UL). We will use the same convention throughout this article, especially in Section 3.

The properties that we require are summarized in the following lemma:

Lemma 2.13 ([30, Lemma 2]). *Given a subset $S \subseteq [\Delta]^2$, we can construct in polynomial time a planar gadget G_S with poly(Δ) unweighted edges and vertices, and an integer D_1 such that the following properties hold:*

(i) For every $(i, j) \in S$, the gadget G_S has a multiway cut of weight D_1 representing (i, j).
(ii) Every multiway cut of G_S has weight at least D_1.
(iii) If a multiway cut of G_S has weight D_1, then it represents some $(i, j) \in S$.

Note that in Reference [30] Marx uses weights to define the gadgets, but as he explains at the end of the introduction, the weights are polynomially large integers and thus can be emulated with parallel unweighted edges.

In the following, we will also use the dual of the graph G_S as one of our gadgets, yielding a ***dual cross gadget*** G_S^* (see Figure 1). Its properties mirror exactly the ones of cross gadgets in a dual setting. In the dual setting, the gadget G_S^* still has the form of a planar graph embedded on a disk D, with $4\Delta + 8$ distinguished faces incident to its boundary, which are, in clockwise order, denoted by

$$UL^*, u_1^*, \dots, u_{\Delta+1}^*, UR^*, r_1^*, \dots, r_{\Delta+1}^*, DR^*, d_{\Delta+1}^*, \dots d_1^*, DL^*, \ell_{\Delta+1}^*, \dots, \ell_1^*.$$

In G_S^*, the vertices dual to boundary faces of G_S lie on the boundary on the disk instead of the interior, see Figure 1, right. As above, the boundary of the disk intersects the graph G_S^* precisely in the distinguished vertices.

A dual multiway cut is a set of edges M^* of G_S^* such that cutting the disk D along M^* yields at least four connected components, and the four terminal faces end up in distinct components. We say that a dual multiway cut M^* ***represents*** a pair $(i, j) \in [\Delta]^2$ if cutting the disk D along M^* yields exactly four connected components that partition the distinguished faces into the following classes:

$$\{UL^*, u_1^*, \dots, u_j^*, \ell_1^*, \dots, \ell_i^*\} \qquad \{UR^*, u_{j+1}^*, \dots u_{\Delta+1}^*, r_1^*, \dots, r_i^*\}$$
$$\{DL^*, d_1^*, \dots d_j^*, \ell_{i+1}^*, \dots, \ell_{\Delta+1}^*\} \qquad \{DR^*, d_{j+1}^*, \dots, d_{\Delta+1}^*, r_{i+1}^*, \dots r_{\Delta+1}^*\}$$

For convenience, we restate the content of Lemma 2.13 in the dual setting in a separate lemma. By duality, its proof follows directly from Lemma 2.13.

Lemma 2.14 ([30, Dual version of Lemma 2]). *Given a subset $S \subseteq [\Delta]^2$, we can construct in polynomial time a planar gadget G_S^* with poly(Δ) unweighted edges and vertices, and an integer D_1 such that the following properties hold:*

(i) For every $(i,j) \in S$, the gadget G_S^* has a dual multiway cut of weight D_1 representing (i,j).
(ii) Every dual multiway cut of G_S^* has weight at least D_1.
(iii) If a dual multiway cut of G_S^* has weight D_1, then it represents some $(i,j) \in S$.

3 THE 4-REGULAR GRAPH TILING PROBLEM

We introduce the problem 4-Regular Graph Tiling, which will be used as a basis to prove the reductions involved in Theorems 1.1 and 1.3.

4-Regular Graph Tiling

Input: Positive integers k, Δ; a four-regular graph Γ on k vertices where the edges are labeled by U, D, L, R in a way that each vertex is incident to exactly one of each label; for each vertex v, a set $S_v \subseteq [\Delta] \times [\Delta]$.

Output: For each vertex v, a value $s_v \in S_v$ such that if $s_v = (i,j)$,

(1) the first coordinates of $s_{L(v)}$ and $s_{R(v)}$ are both i, and
(2) the second coordinates of $s_{U(v)}$ and $s_{D(v)}$ are both j,

where $U(v)$, $D(v)$, $L(v)$, and $R(v)$ denote the vertex of the graph Γ connected to v via an edge labeled respectively by U, D, L, and R.

We call the two conditions above the ***compatibility conditions*** of the 4-Regular Graph Tiling instance. The graph in the input is allowed to have parallel edges. It is easy to see that the Grid Tiling problem [28] is a special case of 4-Regular Graph Tiling.

In this section, we prove a larger lower bound for this more general problem: We prove a $\Delta^{\Omega(k/\log k)}$ lower bound, conditionally to the ETH, for 4-Regular Graph Tiling, even when the problem is restricted to bipartite instances and when fixing k. We also show that it is W[1]-hard when parameterized by the integer k (even for bipartite instances). Precisely:

Theorem 3.1.

(1) The 4-Regular Graph Tiling problem restricted to instances whose underlying graph is bipartite is W[1]-hard parameterized by the integer k.
(2) Assuming the ETH, there exists a universal constant α_{GT} such that for any fixed integer $k \geq 2$, there is no algorithm that decides all the 4-Regular Graph Tiling instances whose underlying graph is bipartite and has at most k vertices, in time $O(\Delta^{\alpha_{GT} \cdot k/\log k})$.

The analogous result for Grid Tiling by Marx [28] embeds the k-Clique problem in a $k \times k$ grid. Here we start from a hardness result for 4-regular binary CSPs that follows from Theorem 2.7 and directly represent the problem as a 4-Regular Graph Tiling instance by locally replacing each variable and each binary constraint in an appropriate way.

Proof. In the proof, we will use the well-known fact that a d-regular bipartite graph G can be properly edge-colored with d colors. This is proved by induction on d: The case $d = 0$ is trivial; in general, take a perfect matching of G, which exists by Hall's marriage theorem; color the edges with color d; the subgraph of G made of the uncolored edges satisfies the induction hypothesis with $d-1$, so it admits a proper edge-coloring with $d-1$ colors; thus G has a proper edge-coloring with d colors. This also implies that computing such a proper edge-coloring takes polynomial time.

The proof of the theorem proceeds by a reduction from the binary CSP instances involved in Theorem 2.6 and Corollary 2.12. Starting from a binary CSP instance $I = (V, D, C)$ whose primal graph is P, a 4-regular bipartite graph, we define an instance of 4-Regular Graph Tiling, $(k, \Delta, \Gamma, \{S_i\})$, in the following way (see Figure 2):

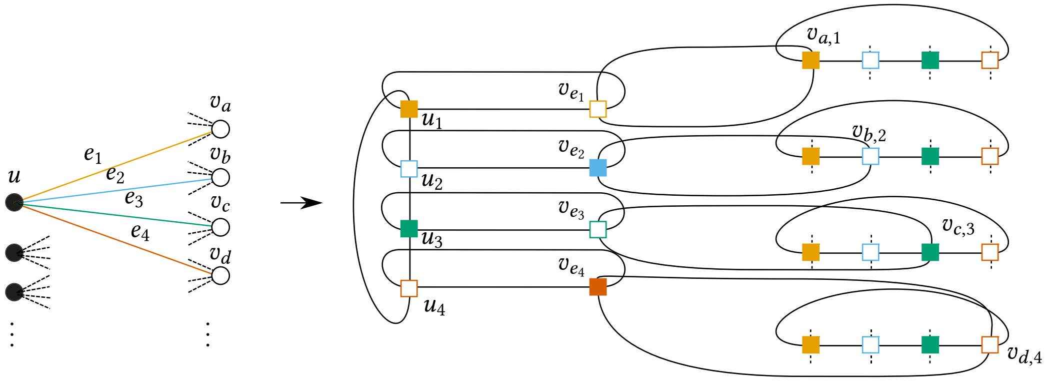

Fig. 2. The reduction in the proof of Theorem 3.1. The bipartition on both sides are represented by hollow/solid vertices. The colors represent the 4-coloring of the edges, and the labels of the edges are suggested by their orientation, i.e., edges entering vertices vertically are labeled U or D, while edges entering vertices horizontally are labeled L or R.

(1) We set $\Delta = |D|$ and $k = 6|V|$.
(2) We find a proper edge coloring of P with 4 colors, as indicated above.
(3) Denoting by V_1 and V_2 the two subsets of vertices of P corresponding to the bipartition of P, for each vertex u of V_1, we create four vertices u_1, u_2, u_3, u_4 in Γ that we connect in a cycle in this order using two U and two D edges (e.g., u_1u_2 and u_3u_4 are U edges and u_2u_3 and u_4u_1 are D edges). Similarly, for each vertex v of V_2, we create four vertices v_1, v_2, v_3, v_4 in Γ that we connect in a cycle in this order using two R and two L edges.
(4) For each edge $e = uv$ labeled with a color i, where $u \in V_1$ and $v \in V_2$, we create one vertex v_e in Γ, which is connected to u_i via two edges, one labeled R and one labeled L, and to v_i via two edges, one labeled U and one labeled D.
(5) For each vertex u_i or v_i of Γ coming from a vertex of P, the corresponding subset S_{u_i} or S_{v_i} is set to be $\text{Diag}([\Delta]) := \{(x,x) \mid x \in [\Delta]\}$.
(6) For each vertex v_e of Γ coming from an edge $e = uv$ of P, where $u \in V_1$ and $v \in V_2$, the corresponding subset S_e is set to be the relation corresponding to e.

We first prove that the graph Γ is bipartite. For this purpose, let W_1 be the set of vertices $\{u_i \mid u \in V_1, i \in \{1,3\}\} \cup \{v_i \mid v \in V_2, i \in \{1,3\}\}$, together with the vertices v_e such that e is labeled by an even color. Let W_2 be the vertices of Γ not in W_1. By construction, each edge in Γ connects a vertex with W_1 with a vertex in W_2, so Γ is indeed bipartite.

We claim that this instance of 4-Regular Graph Tiling is satisfiable if and only if I is satisfiable. Indeed, if I is satisfiable, then the truth assignment f for I can be used to find the values for the s_i in the following way. If $u \in V_1$ and $i \in [4]$, then the value of s_{u_i} is chosen to be $(f(u), f(u))$. Similarly, if $v \in V_2$ and $i \in [4]$, then the value of s_{v_i} is chosen to be $(f(v), f(v))$. For a vertex v_e of Γ coming from an edge $e = uv$ of P where $u \in V_1$ and $v \in V_2$, the value of s_e can be chosen to be $(f(u), f(v))$. The compatibility conditions are trivially fulfilled. In the other direction, the values s_{v_i} for the four vertices of Γ coming from a vertex v of P are identical and of the form (x,x). Choosing x as the truth assignment for v in I yields a solution to the CSP I.

We thus have a reduction from binary CSP, restricted to instances I whose primal graph P has $|V|$ vertices, is four-regular and is bipartite, to instances of 4-Regular Graph Tiling on a bipartite graph Γ with $6|V|$ vertices. This reduction takes time $O((|V||D|)^d)$ for some universal constant $d \geq 1$. Combined with the first item of Corollary 2.12 this proves the first item of the theorem.

For the second item, let $\alpha_{\mathrm{GT}} = \alpha_{\mathrm{REGCSP}}/6$ and let us assume that we have an algorithm $\mathbb{A}$ deciding the 4-Regular Graph Tiling bipartite instances with at most k vertices in time $O(\Delta^{\alpha_{\mathrm{GT}} \cdot k/\log k})$ (where, in the $O()$ notation, we consider k to be fixed). For any four-regular and bipartite graph with at most $k/6$ vertices, the graph obtained from it in the reduction above has at most k vertices. Using this reduction and algorithm $\mathbb{A}$, we could therefore decide any binary CSP instance (V, D, C) whose primal graph is four-regular, bipartite, and has at most $k/6 \geq 2$ vertices in time $O(|D|^d + |D|^{\alpha_{\mathrm{GT}} \cdot k/\log k})$ (for fixed k). This is $O(|D|^{\alpha_{\mathrm{REGCSP}} \cdot (k/6)/\log(k/6)})$ if k is larger than some universal constant $\bar{k}$ (that depends only on d and α_{REGCSP}), and thus, by Corollary 2.12, the ETH does not hold.

This means that that the second item is proved for k at least $\bar{k}$. The remaining cases follow from the trivial linear lower bound, as in Remark 1.2 (note that because $k \geq 2$, there are such instances of 4-Regular Graph Tiling). □

Remark. It might seem more natural to use a definition of 4-Regular Graph Tiling where *half-edges* are labeled by U, D, L, and R, so that every edge contains either U and D, or L and R labels. This fits more the intuition that the top side of a vertex should be attached to the bottom side of the next vertex. It follows from roughly the same proof that the same hardness result also holds for that variant. However, it seems that both the bipartiteness and the unusual labeling are required for the reduction in Section 4.

4 MULTIWAY CUT WITH FOUR TERMINALS

In this section, we prove the following proposition, which will yield Theorem 1.3 in the regime where the genus dominates the number of terminals. The other case will be handled in Section 6.

Proposition 4.1.

(1) *The Multiway Cut problem when restricted to instances (G, T, λ) in which $|T| = 4$ and G is a graph embeddable on the surface of genus g is W[1]-hard parameterized by g.*
(2) *Assuming the ETH, there exists a universal constant α_{MC1} such that for any fixed integer $g \geq 0$, there is no algorithm that decides all the Multiway Cut instances (G, T, λ) for which $|T| = 4$ and G has n vertices and edges and is embeddable on the surface of genus g, in time $O(n^{\alpha_{\mathrm{MC1}} \cdot (g+1)/\log(g+2)})$.*

Proof. The idea is to reduce 4-Regular Graph Tiling instances of Theorem 3.1 to the instances of Multiway Cut specified by the proposition. Consider an instance $(k, \Delta, \Gamma, \{S_i\})$ of 4-Regular Graph Tiling where the underlying graph Γ is bipartite. In polynomial time, we transform it into an equivalent instance (G, T, λ) of Multiway Cut as follows.

(1) For each vertex v of Γ, we create a cross gadget $G_S(v)$ such that the set $S \subseteq [\Delta]^2$ is chosen to be equal to S_v.
(2) For each edge $e = uv$ of Γ labeled U, we identify the vertices of the U side of the cross gadget $G_S(v)$ to the corresponding vertices of the U side of the cross gadget $G_S(u)$. Similarly for the edges labeled D, R, and L for which the vertices on the D, R, and L sides, respectively, are identified. Note that only vertices, and not edges, are identified.
(3) The four corner vertices UL, UR, DR, and DL of all the cross gadgets are identified in four vertices UL, UR, DR, and DL, where the four terminals are placed.
(4) We let $\lambda := kD_1$ (where D_1 is the integer from Lemma 2.13).

Note that since the sides are consistently matched in this last step, the four terminals remain distinct after this identification. (Of course, this identification may create loops and multiple edges.)

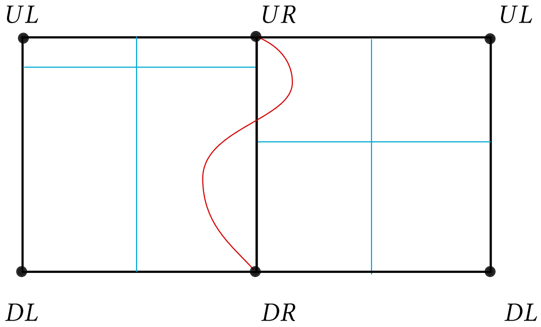

Fig. 3. If the multiway cuts do not match (here represented by their duals), then they do not separate the terminals.

We claim that this instance admits a multiway cut of weight at most λ if and only if the 4-Regular Graph Tiling instance is satisfiable. Assume first that the 4-Regular Graph Tiling instance is satisfiable. For each vertex v of Γ, one can use the value s_v to choose, using Lemma 2.13(1), a multiway cut in $G_S(v)$ representing s_v. Let M be the union of all these sets of edges. We claim that M is a multiway cut separating the four terminals in G. Indeed, after removing the multiway cuts, the four terminals lie in four different components in each of the cross gadgets. It suffices to prove that it remains the case after identifying the four sides. To see this, consider two cross gadgets that have two sides identified, and let w be a vertex on that common side. Then, by the compatibility conditions in the definition of 4-Regular Graph Tiling, w is connected, in the first gadget, to a terminal (UL, UR, DR, or DL) if and only if it is connected to the corresponding terminal in the second gadget. Thus, as desired, M is a multiway cut separating the four terminals in G. Moreover, it has weight at most kD_1, since it is the union of k edge sets of weight at most D_1.

For the other direction, we first observe that if the instance admits a multiway cut of weight at most kD_1, then each of the cross gadgets G_S must admit a multiway cut (otherwise the four terminals would not be disconnected). By Lemma 2.13(2), each of these k multiway cuts has weight exactly D_1. Therefore, by Lemma 2.13(3), each of them represents some $(i, j) \in S$, which will be used as the value s_v for the 4-Regular Graph Tiling instance. Furthermore, we claim that the multiway cuts need to match along identified sides, by which we mean that the two following conditions are satisfied: (1) If a multiway cut represents the pair (i, j), then a multiway cut in a cross gadget adjacent along an edge labeled U or D needs to represent a pair (k, j) for some $k \in [\Delta]$. (2) Similarly, a multiway cut in a cross gadget adjacent along an edge labeled R or L needs to represent a pair (i, ℓ) for some $\ell \in [\Delta]$. Indeed (see Figure 3), if, say, a multiway cut representing the pair (i, j) is connected along an edge labeled R to a multiway cut representing the pair (i', ℓ) for $i' > i$, then vertex (i', Δ), common to both gadgets, is connected to UR by a path inside the first gadget and to DR by a path inside the second gadget, contradicting the fact that we have a multiway cut. Therefore, the compatibility conditions of the 4-Regular Graph Tiling instance are satisfied.

We will prove the following claim.

Claim 4.2. *The genus of the graph G is $O(k)$.*

Proof. We prove here that the genus of the graph G is $O(k)$. For this, the fact that Γ is bipartite turns out to be crucial. For Step 1 above, let us embed the cross gadgets corresponding to Γ in the plane. Let $V_1 \cup V_2$ be the bipartition of the vertices of Γ. We embed the cross gadgets corresponding to V_1 in the plane with the natural orientation (U, R, D, L in clockwise order), and the cross gadgets corresponding to V_2 with the opposite orientation. Second, let us connect the vertices on

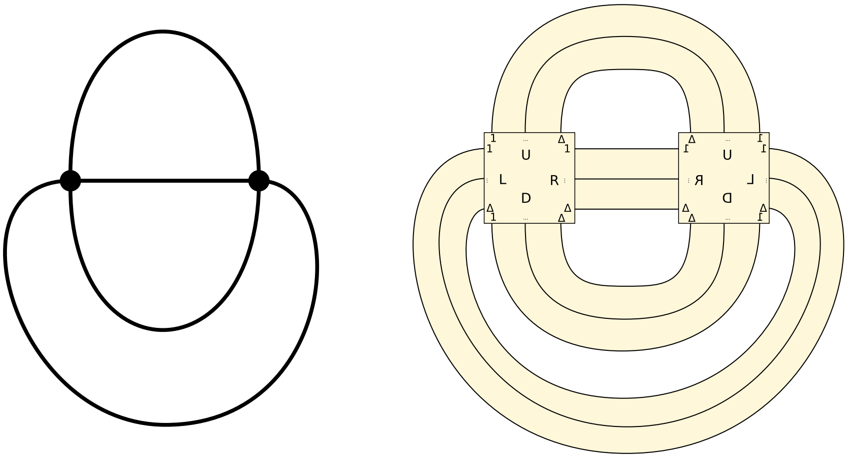

Fig. 4. Left: A bipartite four-valent graph Γ with two vertices. Right: The construction of the embedding of G. The orientation of each gadget of G, corresponding to a vertex v, is chosen according to the side of the bipartition vertex v lies in. This allows to connect pairs of vertices on the boundary of each gadget with the same indices.

the sides of the cross gadgets as in Step 2 above; but for now, just for clarity of exposition, instead of identifying pairs of vertices, let us connect each pair by a new edge. We can add these n new edges corresponding to a single edge of Γ by putting them on a ribbon connecting the sides of the cross gadgets (see Figure 4). We emphasize that because of the orientation chosen to embed the gadgets corresponding to V_1 and V_2, the ribbons are drawn "flat" in the plane (though possibly with some overlapping between them), and the vertices in one cross gadget are connected to the corresponding vertices in the other cross gadget (for example, in the case of an edge labeled R, vertex r_i in the first gadget is connected to vertex r_i in the second gadget). Thus, since we started with a graph embedded on the plane and added at most $2k$ "flat" ribbons (because Γ is four-regular), we obtain a graph embedded on an orientable surface with genus at most $2k$ (without boundary, after attaching disks to each boundary component). We now contract every newly added edge, which can only decrease the genus. For Step 3 above, the graph G is obtained by identifying four groups of at most k vertices of the previous graph (the terminals UL of all cross gadgets, and similarly for UR, DL, and DR) into four vertices; these vertex identifications increase the genus by $O(k)$. (To see this, we can for example add $O(k)$ edges to connect in a linear way all the vertices to be identified, which increases the genus by $O(k)$, and then contract these new edges.) This proves that G is embeddable on a surface of genus $O(k)$. □

To summarize: For some universal constants $c, d \geq 1$, we can transform in time $O((k\Delta)^d)$ any instance $(k, \Delta, \Gamma, \{S_i\})$ of 4-Regular Graph Tiling where Γ is bipartite into an equivalent instance of Multiway Cut with four terminals and whose graph has $O((k\Delta)^d)$ vertices and edges and is embeddable on a surface of genus at most ck. Combined with Theorem 3.1(1), this proves the first item.

Let us now consider the second item. Let $\alpha_{\mathrm{MC1}} = \alpha_{\mathrm{GT}}/c(d+1)$ and assume that for some fixed g there is an algorithm $\mathbb{A}$ that decides all the Multiway Cut instances (G, T, λ) for which G has n vertices and edges and is embeddable on the surface of genus g and $|T| = 4$ in time $O(n^{\alpha_{\mathrm{MC1}} \cdot (g+1)/\log(g+2)})$. Let $k' := \lfloor (g+2)/c \rfloor$ be fixed, and consider an instance $(k, \Delta, \Gamma, \{S_i\})$ of 4-Regular Graph Tiling whose underlying graph is bipartite and has $k \leq k'$ vertices. Using algorithm $\mathbb{A}$ and the above reduction, we can decide this instance in time $O(\Delta^d) + (O(\Delta^d))^{\alpha_{\mathrm{MC1}} \cdot c(k'+1)/\log(ck')}$ (for fixed k'). If g is larger than some universal constant $\bar{g}$ (and thus k'

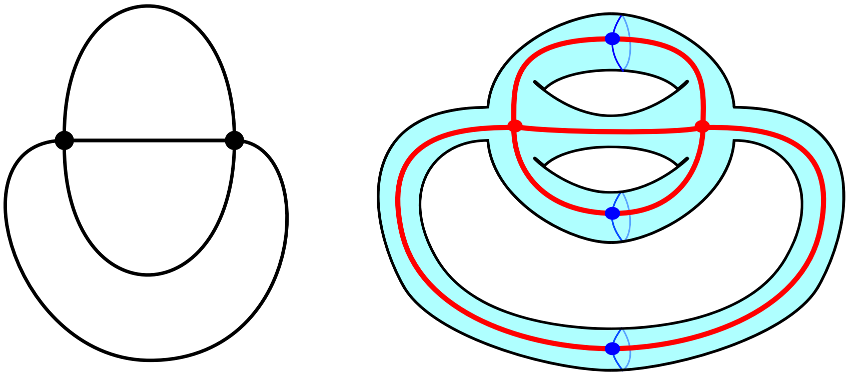

Fig. 5. Left: A bipartite four-valent graph Γ with two vertices. Right: The resulting graph Γ′. The graph Γ is in thick lines, and the edges forming the complement of a spanning tree are split with a new vertex, to which a loop (in thin lines) is attached.

is also large enough), then this is $O(\Delta^{\alpha_{\mathrm{GT}} k'/\log k'})$. Theorem 3.1(2) then implies that the ETH does not hold.

This means that if $g \geq \bar{g}$, then the second item is proved. The remaining cases follow from the trivial linear lower bound, as in Remark 1.2. □

5 SHORTEST CUT GRAPH

In this section, we prove Theorem 1.1 on the hardness of the Shortest Cut Graph problem.

Proof of Theorem 1.1. The idea is to reduce 4-Regular Graph Tiling instances of Theorem 3.1 to instances of Shortest Cut Graph.

Let Γ be a bipartite four-regular graph with k vertices. From Γ, we build an orientable surface without boundary $\mathcal{S}$ as follows (see Figure 5): We build one cylindrical tube for each edge of Γ and one sphere minus four disks for each vertex of Γ, attaching them in the natural way to obtain an orientable surface.

We claim that this surface has genus $k+1$. To see this, let T be a spanning tree of Γ; it has $k-1$ edges, and $E(\Gamma) \setminus T$ has $k+1$ edges. For each edge $e \in E(\Gamma) \setminus T$ in turn, let us remove from $\mathcal{S}$ a cylindrical tube corresponding to e, and attach a disk to each of the two resulting boundary components. Each of these $k+1$ operations decreases the genus by one (as can be seen formally using Euler's formula). We eventually obtain a sphere. This proves the claim.

Moreover, the graph Γ is naturally embedded in $\mathcal{S}$, though not cellularly. To have a cellular embedding, and actually a cut graph, we transform Γ as follows. Let Γ′ be the graph obtained from Γ by subdividing each edge not in T into two edges and adding a loop in the middle vertex. Now, embed Γ′ into $\mathcal{S}$ in the natural way: Starting from the embedding of Γ into $\mathcal{S}$, put each middle vertex on the corresponding cylindrical tube of $\mathcal{S}$, and make the corresponding loop go around the tube. The resulting embedding of Γ′ is a cut graph of $\mathcal{S}$. Indeed, it has a single face, because we only add loops in the middle of edges not in the spanning tree T; moreover, it has $2k+1$ vertices and $4k+2$ edges (being four-regular), so its unique face is a disk, by Euler's formula.

Let $V_1 \cup V_2$ be the bipartition of the vertices of Γ. We note that the above construction is possible while enforcing an arbitrary cyclic ordering of the edges incident to each vertex of Γ; we do it in a way that the cyclic ordering of the edges around each vertex in V_1 is the standard one (U, R, D, L in clockwise order), while the cyclic ordering around each vertex in V_2 is reversed (U, L, D, R in clockwise order). We now build a graph G embedded on the same surface $\mathcal{S}$, obtained by replacing each vertex of Γ' with a dual cross gadget and by identifying vertices on the corresponding sides of adjacent gadgets. In detail:

(1) For each vertex v of Γ, we create a dual cross gadget $G^*_S(v)$. To define this gadget, we only have to specify the corresponding integer Δ together with the subset S of $[\Delta]^2$; we choose $\Delta := n$ and $S := S_v$. We embed that dual cross gadget with the same orientation as the corresponding vertex of Γ'.

(2) For each edge $e = uv$ of T, we identify the vertices (not the edges) on the side of $G^*_S(u)$ corresponding to the label of e to the vertices on the same side of $G^*_S(v)$. By the choice of the rotation systems, and for the same reason as in Figure 4, this identifies the vertices in the gadget associated with u to the corresponding vertices in the gadget associated with v; for example, if the label of edge e is R, then the vertex r_i of the first gadget is associated to vertex r_i of the second gadget.

(3) For an edge $e = uv$ of Γ not in T, we use another dual cross gadget $G^*_S(e)$, for which we choose S to be the unconstrained relation $S = [n]^2$. We put that gadget on the vertex of Γ' that splits edge e. We identify the vertices on the side of $G^*_S(u)$ corresponding to the label of e to the vertices of the same side of $G^*_S(e)$, and similarly the vertices on the side of $G^*_S(v)$ corresponding to the label of e to the vertices on the opposite side of $G^*_S(e)$. The two sets of vertices on the opposite sides of $G^*_S(e)$ that are not yet identified are identified to each other.

As a result of the identifications, some corners of the dual cross gadgets might become identified. Unlike in the reductions for Multiway Cut, this is not a problem here. The following claim shows that the reduction works as expected.

Claim 5.1. *The 4-Regular Graph Tiling instance on Γ is satisfiable if and only if the embedded graph G admits a cut graph C of weight at most $(2k + 1)D_1$.*

Proof (Recall that Γ' has $2k + 1$ vertices). Let us first assume that the 4-Regular Graph Tiling instance is satisfiable. For each vertex v of Γ, one can use the value s_v to choose, using Lemma 2.14(1), a dual multiway cut in $G^*_S(v)$ representing s_v. For $s_v = (i, j)$, such a dual multiway cut contains a path connecting the vertex between the ith and $(i + 1)$st face of $G^*_S(v)$ on the side L to the vertex between ith and $(i + 1)$st face of $G^*_S(v)$ on the side R, and likewise there is a path connecting the U and D side between the jth and $(j + 1)$st faces. Since the (s_v) satisfy the compatibility conditions, such paths in two adjacent dual cross gadgets must match on the common boundary. Thus the union of the edges of these dual multiway cuts contains a cut graph of $\mathcal{S}$; indeed, it contains a subdivision of the graph Γ' described above (possibly after transforming a degree-four vertex into two degree-three vertices connected by an edge), which is a cut graph. Furthermore, this union has weight $(2k + 1)D_1$, since each of the dual multiway cuts has weight D_1. Thus, some subgraph of it, of weight at most $(2k + 1)D_1$, is a cut graph.

The reverse direction requires more effort. First, let us call a cut graph *reduced* if it has no degree-one vertex; Γ' is clearly reduced. We will use the following fact: Every simple closed curve γ crossing some reduced cut graph exactly once is non-contractible. Indeed, if γ is contractible, then it bounds a disk; since the cut graph intersects the boundary of the disk exactly once, the part of the cut graph inside the disk is a tree, and thus has at least two degree-one vertices, one of which is not on γ; this is a contradiction.

Assume that the embedded graph G admits a cut graph C of weight at most $(2k + 1)D_1$. Note that the edge set of G is the disjoint union of the edges of the dual cross gadgets, and one can naturally talk about the restriction of (the set of edges of) C to a given dual cross gadget (before the identification of the vertices of the dual cross gadgets). We first prove that the restriction of C

to each of the dual cross gadgets G^*_S contains a dual multiway cut. Indeed, assume that there is a path p, in the disk corresponding to gadget G^*_S, that connects two terminal faces and does not cross C. Let q be a path with the same endpoints as p and that does not enter the gadgets; such a path exists because the union of the gadgets is a thickened version of the graph Γ', which is a cut graph, and which thus has a single face, and because the terminal faces are on the boundary of the dual cross gadget. The closed curve γ that is the concatenation of p and q is contractible, because it does not cross C. However, we can slightly modify Γ' in the disk of the gadget G^*_S to obtain a reduced cut graph that crosses p, and thus also γ, exactly once (which, by the previous paragraph, implies that γ is contractible, and thus the contradiction). To prove this fact, there are two cases. If p connects two "neighboring" terminals in G^*_S, e.g., UL and DL, then we locally homotope the part of Γ' inside the disk of G^*_S to the R side of that disk, except for the edge that crosses the L side, and we draw that latter edge in a way that it crosses p exactly once. If p connects two "opposite" terminals in G^*_S, e.g., UL and DR, then we first replace the four-valent vertex of Γ' in the gadget with two three-valent vertices v_1 and v_2, in a way that v_1 is connected to the L and D sides and to v_2, and similarly v_2 is connected to the U and R sides and to v_1. We put v_1 in the DL corner, v_2 in the UR corner, and connect them by an edge crossing p exactly once. Thus, in both cases, we have a (clearly reduced) cut graph that crosses p, and thus γ, exactly once, as desired.

Thus, the restriction of C to each dual cross gadget $G^*_S(u)$ or $G^*_S(e)$ is a dual multiway cut representing some $(i,j) \in S$, by Lemma 2.13(2) and (3). Furthermore, the dual multiway cuts must match on the boundaries, i.e., if a dual multiway cut represents the pair (i,j), then the multiway cut in a cross gadget adjacent along an edge labeled U or D needs to represent a pair (k,j) for some $k \in [\Delta]$, and similarly the dual multiway cut in a cross gadget adjacent along an edge labeled R or L needs to represent a pair (i,ℓ) for some $\ell \in [\Delta]$. Indeed, otherwise, there is a path connecting two terminal faces, as pictured in Figure 3, and by a similar argument as above, this path can be completed into a non-contractible cycle not crossing C, which is a contradiction.

Since the restriction of C to each dual cross gadget is a dual multiway cut that has to represent some $(i,j) \in S$, we use (i,j) as the value of s_v for the 4-Regular Graph Tiling instance. The compatibility conditions follow from the fact that the dual multiway cuts match on the boundaries; thus, the 4-Regular Graph Tiling instance is satisfiable. □

To summarize: For some universal constant $d \geq 1$, we can transform in time $(k\Delta)^d$ any instance $(k, \Delta, \Gamma, \{S_i\})$ of 4-Regular Graph Tiling where Γ is bipartite into an equivalent instance of Shortest Cut Graph whose graph has at $O((k\Delta)^d)$ vertices and edges, embedded on a surface of genus $k+1$. Combined with Theorem 3.1(1), this proves the first item of the theorem.

Let us now consider the second item. Let $\alpha_{\mathrm{CG}} = \alpha_{\mathrm{GT}}/(d+1)$ and assume that, for some fixed g, there is an algorithm $\mathbb{A}$ that decides all the Shortest Cut Graph instances embedded on a surface of genus at most g in time $O(n^{\alpha_{\mathrm{CG}} \cdot (g+1)/\log(g+2)})$. Let $k' = g-1$, and consider an instance $(k, \Delta, \Gamma, \{S_i\})$ of 4-Regular Graph Tiling whose underlying graph is bipartite and has $k \leq k'$ vertices. Using algorithm $\mathbb{A}$ and the above reduction, we can decide this instance in time $O(\Delta^d) + (\Delta^d)^{\alpha_{\mathrm{CG}} \cdot (k'+2)/\log(k'+3)}$ (for fixed g, and thus fixed k'). If g is larger than a certain universal constant $\bar{g}$, then this is $O(\Delta^{\alpha_{\mathrm{GT}} k'/\log k'})$. Theorem 3.1(2) then implies that the ETH does not hold.

This means that there exists an integer $\bar{g}$ such that, if $g \geq \bar{g}$, then the theorem is proved. The remaining cases follow from the trivial linear lower bound, as in Remark 1.2. □

6 MULTIWAY CUT WITH A LARGE NUMBER OF TERMINALS

The goal of this section is to prove the following proposition, which will yield Theorem 1.3 in the regime where the number of terminals dominates the genus.

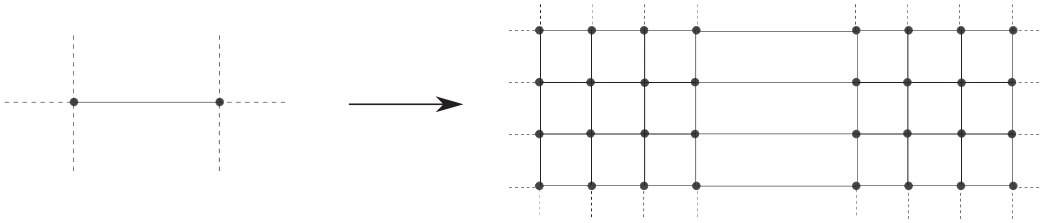

Fig. 6. The construction of G^{δ} for $\delta = 4$.

Proposition 6.1. *Assuming the ETH, there exists a universal constant α_{MC2} such that for any fixed choice of integers $g \geq 0$ and $t \geq 48(g+1)$, there is no algorithm that decides all the* Multiway Cut *instances (G, T, λ) for which G is embeddable on the surface of genus g and such that $|T| \leq t$, in time $O(n^{\alpha_{\text{MC2}}\sqrt{gt+t}/\log(g+t)})$.*

To prove this proposition, we will blow up expander graphs by replacing their vertices by grids, a construction similar to the one in Gilbert, Hutchinson, and Tarjan [15]. Given a four-regular graph cellularly embedded on a surface $\mathcal{S}$, we define G^{δ} in the following way (see Figure 6). We replace each vertex of G with a $(\delta \times \delta)$-grid. The boundary of the grid is thus made of $4\delta - 4$ vertices, which we divide into four contiguous segments of δ vertices starting from a corner, in clockwise order along the boundary of the grid (corners appear in two segments). For each edge e of G between two adjacent vertices u and v, we blow up e in the following way: we place an edge between the ith vertex of a segment of u and the $(\delta - i + 1)$st vertex of a segment of v. The segments are selected in such a way that the cyclic ordering of the four blown-up half-edges around a grid replacing a vertex u corresponds to the cyclic ordering of the four half-edges around u that is prescribed by the cellular embedding. This new graph G^{δ} is also cellularly embedded on $\mathcal{S}$, since the embedding of G can be "thickened" to an embedding of G^{δ}. Moreover, G^{δ} clearly has $\delta^2 \cdot |V(G)|$ vertices.

The properties of G^{δ} that we are interested in are summarized in the following lemma.

Lemma 6.2. *Let G be a four-regular graph cellularly embedded on a surface $\mathcal{S}$, such that $\lambda(G)/4 < c < 1$ for some universal constant c, and let δ be a positive integer. The graph G^{δ} can also be cellularly embedded on $\mathcal{S}$, has $\delta^2 \cdot |V(G)|$ vertices and treewidth $\Omega(\delta \cdot |V(G)|)$.*

To prove the previous lemma, we will use the following auxiliary, probably folklore, result.

Lemma 6.3. *Let H be the $(\delta_1 \times \delta_2)$-grid where $\delta_1 \leq \delta_2$, and let $\alpha < 1$ be given. There exists $\beta > 0$, depending only on α and δ_2/δ_1, such that any α-separator of H has more than $\beta\delta_1$ vertices.*

Proof. So let H have δ_1 rows and δ_2 columns. Let β be chosen such that $0 < \beta < \min\{1, (1-\alpha)\delta_2/\delta_1\}$. Let C be a subset of vertices of H of size at most $\beta\delta_1$. Thus, at least one row and $\delta_2 - \beta\delta_1$ columns do not intersect C, and there is a connected component of $H - C$ of size at least $\delta_1(\delta_2 - \beta\delta_1) > \alpha\delta_1\delta_2$, and C is not an α-separator. □

Proof of Lemma 6.2. Following the construction, and by Lemma 2.3, there just remains to prove that $\text{tw}(G^{\delta}) = \Omega(\delta \cdot \text{tw}(G))$. The approach mimicks the lower bound proof of Gilbert, Hutchinson, and Tarjan [15] and adapts it to the case of four-regular graphs. Their initial proof is for three-regular graphs. Let $G' := G^{\delta}$. For each vertex v of the original graph G, we denote by $gr(u)$ the associated $(\delta \times \delta)$-grid in G'.

Most of the proof consists in proving that if G' has a 1/4-separator of size k, then G has a 1/2-separator of size $O(k/\delta)$. For this purpose, we first note, by Lemma 6.3, the existence of a universal constant $\beta > 0$ such that:

- for every vertex u of G, every $1/2$-separator of $gr(u)$ has more than $\beta\delta$ vertices, and
- for every edge uv of G, every $3/4$-separator of $G'[gr(u) \cup gr(v)]$ has more than $2\beta\delta$ vertices.

Let C' be a $1/4$-separator for G'. Let C be the set of vertices u of G such that $gr(u)$ contains more than $\beta\delta$ vertices of C'. We will prove that C is a $1/2$-separator of G of size $O(|C'|/\delta)$.

Let u be a vertex of $G - C$. By definition of β and C, we have that $gr(u) \cap C'$ is not a $1/2$-separator of $gr(u)$, and thus the strict majority of the vertices of $gr(u)$ lie in a single connected component of $G' - C'$. Thus, let $A'_1, \ldots, A'_r$ be the connected components of $G' - C'$; for $i = 1, \ldots, r$, let A_i be the set of vertices u of $G - C$ such that the strict majority of the vertices of $gr(u)$ lie in A'_i. We have that $C, A_1, \ldots, A_r$ form a partition of the vertices of G.

We first argue that C separates all pairs A_i, A_j, $i \neq j$. Assume to the contrary that there are adjacent vertices u and v of G such that $u \in A_i$ and $v \in A_j$. By definition of A_i and A_j, neither $gr(u)$ nor $gr(v)$ contains more than $\beta\delta$ vertices of C'. Thus $G'[gr(u) \cup gr(v)]$ contains at most $2\beta\delta$ vertices of C'. Thus, by definition of β, a fraction larger than $3/4$ of its vertices of $G'[gr(u) \cup gr(v)]$, namely more than $3\delta^2/2$ vertices, are in a single component of $G' - C'$. It follows that both $gr(u)$ and $gr(v)$ have more than $\delta^2/2$ vertices in that component, so they are both in the same set A'_i, a contradiction.

We then show that each A_i is small. Let n be the number of vertices of G. Recall that (i) $|V(A'_i)| \leq \delta^2 n/4$, because C' is a $1/4$-separator, and (ii) $|V(A'_i)| > |V(A_i)| \cdot \delta^2/2$, by definition of A_i. These two inequalities imply $|V(A_i)| \leq n/2$.

Hence C is a $1/2$-separator. Finally, by definition of C, we have $|C'| > \beta\delta|C|$. Thus $|C| = O(|C'|/\delta)$.

We have proved that if G' has a $1/4$-separator of size k, then G has a $1/2$-separator of size $O(k/\delta)$. However, since $\lambda(G)/4 < c < 1$, by Lemma 2.3, there exists a constant $c' > 0$ such that G has no $1/2$-separator of size at most $c'|V(G)|$. Therefore, there exists a constant $c'' > 0$ such that G' does not have a $1/4$-separator of size at most $c''\delta|V(G)|$. This implies that there exists a constant $c''' > 0$ such that G' has treewidth at least $c'''\delta|V(G)|$. Indeed, otherwise, by Lemma 2.2 we could find a $1/2$-separator of G' of size at most $c'''\delta|V(G)| + 1$, and by applying it again on at most two connected components and taking the union of the separators, for a small enough c''' we would find a $1/4$-separator of G' of size at most $c''\delta|V(G)|$, a contradiction. Therefore G' has treewidth $\Omega(\delta \cdot |V(G)|)$, which concludes the proof. □

The previous considerations are used in the following lemma, which essentially provides a four-regular graph with $\Theta(t)$ vertices, genus at most g, and treewidth $\Theta(\sqrt{gt})$. Intuitively, if we start with an expander G of size g and set $\delta = \sqrt{t/g}$, then G^δ has $\Theta(t)$ vertices and treewidth $\Theta(\sqrt{gt})$.

Lemma 6.4. *There is a universal constant* c_{tw} *such that, for every choice of* $g \geq 0$ *and* $t \geq 48(g+1)$, *there is a four-regular graph* P *embeddable on a surface of genus* g *such that* $|V(P)| \leq t/12$ *and* $tw(P) \geq \max\{2, c_{\mathrm{tw}}\sqrt{gt+t}\}$.

Proof. Consider the family of expanders $\mathcal{H}$ defined in Lemma 2.1 and let $\mu \geq 1$ be the number of vertices of the smallest graph in that family. Then, by Lemma 2.1, for a small-enough universal constant $c > 0$, for any $g \geq \mu/c$, there exists an expander $H \in \mathcal{H}$ such that $c \cdot (g+1) \leq |V(H)| \leq g/2$.

Consider g and t as in the lemma; assume first that indeed $g \geq \mu/c$, and let H be as indicated. Since H is four-regular, it has at most g edges, so its genus is at most g. Let δ be $\lfloor\sqrt{t/(12|V(H)|)}\rfloor$. Since $t \geq 48(g+1) \geq 96|V(H)|$, we have $\delta \geq 2$ and $\delta = \Theta(\sqrt{t/(g+1)})$. Moreover, let $P := H^\delta$ (this construction depends on a choice of a cellular embedding for H; we choose an arbitrary one on a surface of genus at most g). Like H, the graph P is embeddable on the surface of genus g. By the choice of δ, the four-regular graph P has $\Omega(t)$ vertices, but at most $t/12$ vertices. The graph H

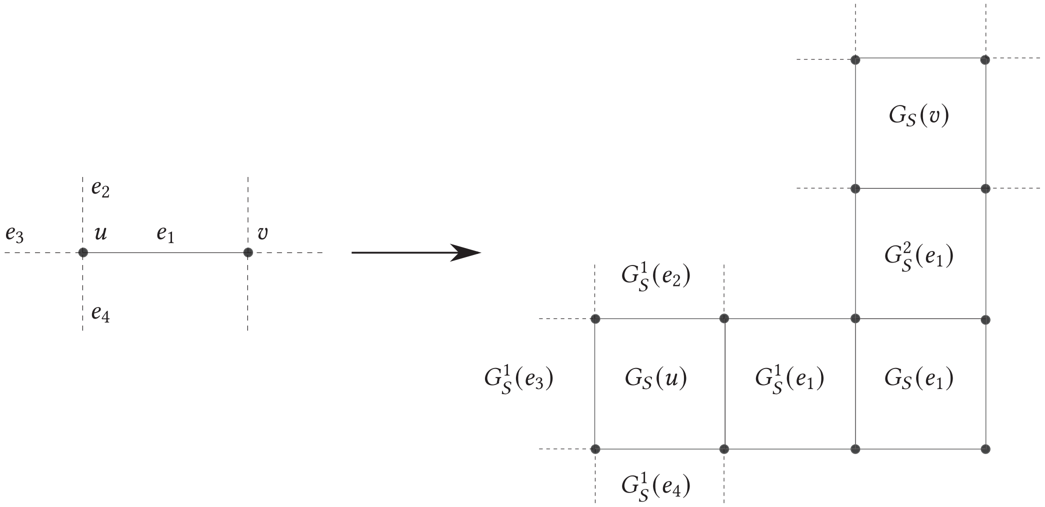

Fig. 7. Reduction from a binary CSP instance to a Multiway Cut instance. The figure displays the gadgets corresponding to an edge $e_1 = uv$ of the binary CSP instance. Note that $G_S^1(e_1)$ is attached to an arbitrary side of $G_S(u)$, and similarly $G_S^2(e_1)$ is attached to an arbitrary side of $G_S(v)$. What matters is that the gadgets adjacent to $G_S^1(e_1)$ (respectively, $G_S^2(e_1)$) are on opposite sides of this gadget; and similarly that the gadgets adjacent to $G_S(e_1)$ are on non-opposite sides of this gadget.

has treewidth $\Omega(g+1)$ by Lemma 2.3, so Lemma 6.2 implies that P has treewidth $\Omega(\delta(g+1)) = \Omega(\sqrt{gt+t})$. Moreover, this treewidth is at least two, because P contains a $\delta \times \delta$-grid. This proves the lemma if $g \geq \mu/c$.

Let now g and t be as in the lemma, with $g < \mu/c$. We let H be the planar graph with one vertex and two loops, and let δ be equal to $\lfloor\sqrt{t/12}\rfloor$. Note that $t \geq 48(g+1) \geq 48$; thus, $\delta \geq 2$. The graph $P := H^\delta$ has $\delta^2 \leq t/12$ vertices; it contains a $\delta \times \delta$-grid; hence, it has treewidth at least δ, which is both at least two and $\Omega(\sqrt{gt+t})$. Moreover, P is planar and thus embeds on a surface of genus at most g. This proves the lemma if $g < \mu/c$. □

Armed with Lemma 6.4, we can now proceed to the proof of Proposition 6.1.

Proof of Proposition 6.1. The reduction is slightly different from the previous ones. It follows the same ideas, with the help of the previous lemmas, but to obtain exactly the lower bound, we will use directly the full strength of Theorem 2.7 and not go through the 4-Regular Graph Tiling problem.

Let P be a four-regular graph cellularly embedded on a surface $\mathcal{S}$ of genus at most g. We will make use of Theorem 2.7 on CSPs whose primal graph is P. Given a binary CSP instance $I = (V, D, C)$ whose primal graph is P, we can transform it in polynomial time into an equivalent instance (G, T, λ) of Multiway Cut where $|T| = 12|V(P)|$ and G is cellularly embedded on $\mathcal{S}$ and has $|V(P)| \cdot \text{poly}(|D|)$ vertices and edges, as follows:

(1) For each vertex v of P, we create a cross gadget $G_S(v)$ where $\Delta = |D|$ and the subset S is $\text{Diag}([\Delta]) := \{(x,x) \mid x \in [\Delta]\}$.
(2) For each edge $e = uv$ of P, we create a cross gadget $G_S(e)$ where $\Delta = |D|$, as well as two cross gadgets $G_S^1(e)$ and $G_S^2(e)$. The cross gadgets $G_S(u)$ and $G_S(v)$ are connected via the three gadgets $G_S^1(e)$, $G_S(e)$ and $G_S^2(e)$ as pictured in Figure 7, i.e., $G_S(u)$ has an arbitrary side identified to the L side of $G_S^1(e)$, of which the R side is identified to the L side of $G_S(e)$,

of which the U side is identified to the D side of $G_S^2(e)$, which is finally identified via its U side to an arbitrary side of $G_S(v)$. This is done in such a way that the cyclic ordering of the four cross gadgets around a cross gadget $G_S(u)$ corresponds to the cyclic ordering of the four edges around u specified by the embedding of P. This way, the new graph G is cellularly embedded on the same surface $\mathcal{S}$.

(3) For each gadget $G_S^1(e)$ or $G_S^2(e)$, the corresponding set S is the unconstrained relation $[\Delta] \times [\Delta]$. (These gadgets are here only to separate the gadgets corresponding to the vertices of P, see below.) For $G_S(e)$, the corresponding set S corresponds to the relation on e, but we need to take care of the orientations. Formally, let ι be the identity permutation on $[\Delta]$ and ρ be the reversing permutation on $[\Delta]$ (namely, $\rho(i) = \Delta + 1 - i$). If $G_S^1(e)$ is attached to $G_S(u)$ via the R or U side of $G_S(u)$, then let $\sigma_e := \iota$; otherwise, let $\sigma_e := \rho$. Thus, σ_e encodes the permutation between the indices of the side of $G_S(u)$ and the side of $G_S(e)$ that are connected through $G_S^1(e)$. Similarly, if $G_S^2(e)$ is attached to $G_S(v)$ via the D or L side of $G_S(v)$, then let $\tau_e := \iota$; otherwise, let $\tau_e := \rho$. Finally, the set S corresponding to gadget $G_S(e)$ contains (i, j) if and only if $(\sigma_e(i), \tau_e(j))$ is in the relation corresponding to e.

(4) We place terminals at every corner of every cross gadget. Let T be this set of terminals; by construction, $|T| = 4(|V(P)| + |E(P)|) = 12|V(P)|$.

(5) We let $\lambda := 7|V(P)|D_1$.

Note that at the end of Step 2, each cross gadget for a vertex has its four sides identified to another gadget, while each of the three cross gadgets for an edge has two sides left unidentified. Therefore, after Step 4, a corner of a cross gadget lies in either two or three distinct cross gadgets, so every cross gadget has at its corners four distinct terminals.

Let us first assume that the CSP instance is satisfiable. For each vertex v of P, one can use the value s_v to choose, using Lemma 2.13(1), a multiway cut in $G_S(v)$ representing (s_v, s_v). Similarly, in the gadgets adjacent to $G_S(v)$, we choose a multiway cut representing (s_v, s_v). Finally, in the gadget $G_S(e)$ where $e = uv$, we choose a multiway cut representing $(\sigma_e(s_u), \tau_e(s_v))$. By construction, taking the union of all these sets of edges forms a multiway cut separating all the terminals in G. Indeed, after removing the multiway cuts, the four terminals of each cross gadget lie in four different components. This remains the case after identifying the sides: The binary constraints of the CSP force the multiway cuts to match along two consecutive sides. The multiway cut has weight exactly λ, since it is the union of $7|V(P)|$ edge sets of weight at most D_1.

Let us prove the other direction. First, if (G, T) admits a multiway cut of weight at most $\lambda = 7|V(P)|D_1$, then each of the cross gadgets G_S must admit a multiway cut (otherwise the four terminals would not be disconnected). Since there are $7|V(P)|$ of those, and by Lemma 2.13(2), each of these multiway cuts has weight exactly D_1. Therefore, by Lemma 2.13(3), each of them has to represent some $(i, j) \in S$. The value on the gadgets $G_S(v)$ are of the form (s_v, s_v); we let s_v be the value for the binary CSP instance for variable v. Furthermore, the multiway cuts need to match along consecutive sides, otherwise, the four terminals are not separated in one of the cross gadgets, as pictured in Figure 3. Therefore, the binary constraints of the CSP instance are satisfied.

To summarize: For some universal constant $d \geq 1$, given a binary CSP instance $I = (V, D, C)$ whose primal graph is a four-regular graph P cellularly embedded on a surface of genus at most g, we obtain in time $O((|V||D|)^d)$ an equivalent instance (G, T, λ) of Multiway Cut where $|T| \leq 12|V|$ and G has $O((|V||D|)^d)$ vertices and edges and is embeddable on a surface of genus g.

Assume that, for some fixed g, and for some fixed but large-enough $t \geq 48(g + 1)$, there is an algorithm $\mathbb{A}$ that decides all the Multiway Cut instances (G, T, λ) for which G is embeddable on a surface of genus g and such that $|T| \leq t$ in time $O(n^{(\alpha_{\mathrm{CSP}} c_{\mathrm{tw}}/d) \cdot \sqrt{g+t}/\log(g+t)})$. Let P be obtained as in Lemma 6.4, and consider a binary CSP instance $I = (V, D, C)$ whose primal graph is P. Since

$|V(P)| \leq t/12$ and t is fixed, we can in constant time compute a cellular embedding of P on a surface of genus at most g. Using algorithm $\mathbb{A}$ and the above reduction, we can decide this instance in time $O(|D|^d) + O(|D|^d)^{(\alpha_{\mathrm{CSP}} c_{\mathrm{tw}}/d)\cdot\sqrt{gt+t}/\log(g+t)}$ (for fixed g and t). Since also $\mathrm{tw}(P) \geq c_{\mathrm{tw}}\sqrt{gt+t}$ and $\log(g+t) \geq \log \mathrm{tw}(P)$ (because P has at most $t/12$ vertices), this is $O(|D|^{\alpha_{\mathrm{CSP}}\cdot \mathrm{tw}(P)/\log \mathrm{tw}(P)})$ provided t is large enough. Finally, by construction, P has treewidth at least two. Hence by Theorem 2.7, the Exponential Time Hypothesis does not hold.

So, there exists an integer $\bar{t}$ such that the following holds: If for some fixed g, and for some fixed but large-enough $t \geq \bar{t}$ and $t \geq 48(g+1)$, there is an algorithm that decides all the Multiway Cut instances (G, T, λ) for which G is embeddable on a surface of genus g and such that $|T| \leq t$ in time $O(n^{(\alpha_{\mathrm{CSP}} c_{\mathrm{tw}}/d)\cdot\sqrt{gt+t}/\log(g+t)})$, then the ETH does not hold.

Let $\alpha_{\mathrm{MC2}} > 0$ be at most $\alpha_{\mathrm{CSP}} c_{\mathrm{tw}}/d$ and such that $\alpha_{\mathrm{MC2}} \cdot \sqrt{gt+t}/\log(g+t) < 1$ for each g and t such that $\bar{t} > t \geq 48(g+1)$. There is a trivial, unconditional $\Omega(n)$ lower bound for Multiway Cut. Thus, for each g and t such that $\bar{t} > t \geq 48(g+1)$, there is no algorithm deciding the Multiway Cut instances on a surface of genus g with t terminals in time $O(n^{(\alpha_{\mathrm{MC2}}\cdot\sqrt{gt+t}/\log(g+t))})$. Finally, if for some g and t such that $t \geq 48(g+1)$, then there is an algorithm deciding the Multiway Cut instances on a surface of genus g with t terminals in time $O(n^{(\alpha_{\mathrm{MC2}}\cdot\sqrt{gt+t}/\log(g+t))})$, then the ETH does not hold. □

7 PROOF OF THEOREM 1.3

Finally, the proof of Theorem 1.3 is obtained by using Proposition 4.1 or Proposition 6.1, depending on the tradeoff between g and t:

Proof of Theorem 1.3. Let $\alpha_{\mathrm{MC}} > 0$. Let us assume that for some fixed choice of integers $g \geq 0$ and $t \geq 4$, there is an algorithm $\mathbb{A}$ that solves the Multiway Cut instances (G, T, λ) for which G is embeddable on the orientable surface of genus g and $|T| \leq t$ in time $O(n^{\alpha_{\mathrm{MC}}\sqrt{gt+g^2+t}/\log(g+t)})$. We distinguish according to two cases for the choice of (g, t).

- If $t < 48(g+1)$, then $\sqrt{gt+g^2+t} \leq \sqrt{97(g+1)^2}$ and thus $\mathbb{A}$ solves, in particular, the Multiway Cut instances where G is embeddable on the surface of genus g and $|T| = 4$, in time $O(n^{\alpha_{\mathrm{MC}}\cdot 10(g+1)/\log(g+2)})$, so Proposition 4.1 implies that the ETH does not hold if $\alpha_{\mathrm{MC}} \leq \alpha_{\mathrm{MC1}}/10$.
- If $t \geq 48(g+1)$, then $\sqrt{gt+g^2+t}/\log(g+t) \leq \sqrt{2(gt+t)}/\log(g+t)$, and thus Proposition 6.1 implies that the ETH does not hold if $\alpha_{\mathrm{MC}} \leq \alpha_{\mathrm{MC2}}/\sqrt{2}$.

Thus, assuming that $\alpha_{\mathrm{MC}} > 0$ is small enough, the existence of $\mathbb{A}$ implies that the ETH does not hold, as desired. □

REFERENCES

[1] Jianer Chen, Iyad A. Kanj, Ljubomir Perković, Eric Sedgwick, and Ge Xia. 2007. Genus characterizes the complexity of certain graph problems: Some tight results. *J. Comput. System Sci.* 73, 6 (2007), 892–907.

[2] Rajesh Hemant Chitnis, Mohammad Taghi Hajiaghayi, and Dániel Marx. 2014. Tight bounds for planar strongly connected steiner subgraph with fixed number of terminals (and extensions). In *Proceedings of the 25th ACM-SIAM Symposium on Discrete Algorithms (SODA'14)*. 1782–1801.

[3] Vincent Cohen-Addad, Éric Colin de Verdière, and Arnaud de Mesmay. 2018. A near-linear approximation scheme for multicuts of embedded graphs with a fixed number of terminals. In *Proceedings of the 29th ACM-SIAM Symposium on Discrete Algorithms (SODA'18)*. 1439–1458.

[4] Vincent Cohen-Addad and Arnaud de Mesmay. 2015. A fixed parameter tractable approximation scheme for the optimal cut graph of a surface. In *Proceedings of the European Symposium on Algorithms (ESA'15)*. Springer, 386–398.

[5] Éric Colin de Verdière. 2017. Multicuts in planar and bounded-genus graphs with bounded number of terminals. *Algorithmica* 78, 4 (2017), 1206–1224.

[6] Éric Colin de Verdière. 2018. Computational topology of graphs on surfaces. In *Handbook of Discrete and Computational Geometry* (3rd ed.), Jacob E. Goodman, Joseph O'Rourke, and Csaba Toth (Eds.). CRC Press LLC, 605–636.

[7] Radu Curticapean and Dániel Marx. 2016. Tight conditional lower bounds for counting perfect matchings on graphs of bounded treewidth, cliquewidth, and genus. In *Proceedings of the 27th ACM–SIAM Symposium on Discrete Algorithms (SODA'16)*. Society for Industrial and Applied Mathematics, 1650–1669.

[8] Marek Cygan, Fedor V. Fomin, Łukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michał Pilipczuk, and Saket Saurabh. 2015. *Parameterized Algorithms*. Vol. 3. Springer.

[9] Elias Dahlhaus, David S. Johnson, Christos H. Papadimitriou, Paul D. Seymour, and Mihalis Yannakakis. 1994. The complexity of multiterminal cuts. *SIAM J. Comput.* 23, 4 (1994), 864–894.

[10] Mark de Berg, Hans L. Bodlaender, Sándor Kisfaludi-Bak, Dániel Marx, and Tom C. van der Zanden. 2018. A framework for ETH-tight algorithms and lower bounds in geometric intersection graphs. In *Proceedings of the 50th ACM Symposium on Theory of Computing (STOC'18)*. 574–586.

[11] Erik D. Demaine, Fedor V. Fomin, Mohammadtaghi Hajiaghayi, and Dimitrios M. Thilikos. 2005. Subexponential parameterized algorithms on bounded-genus graphs and H-minor-free graphs. *J. ACM* 52, 6 (2005), 866–893.

[12] Reinhard Diestel. 2000. *Graph Theory*. Springer-Verlag. Retrieved from http://diestel-graph-theory.com/.

[13] Jeff Erickson and Sariel Har-Peled. 2004. Optimally cutting a surface into a disk. *Discr. Comput. Geometry* 31, 1 (2004), 37–59.

[14] Fedor V. Fomin, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. 2016. Subexponential parameterized algorithms for planar and apex-minor-free graphs via low treewidth pattern covering. In *Proceedings of the IEEE 57th Symposium on Foundations of Computer Science (FOCS'16)*. 515–524. DOI: https://doi.org/10.1109/FOCS.2016.62

[15] John R. Gilbert, Joan P. Hutchinson, and Robert Endre Tarjan. 1984. A separator theorem for graphs of bounded genus. *J. Algor.* 5, 3 (1984), 391–407.

[16] Martin Grohe. 2007. The complexity of homomorphism and constraint satisfaction problems seen from the other side. *J. ACM* 54, 1 (2007), 1.

[17] Martin Grohe and Dániel Marx. 2009. On tree width, bramble size, and expansion. *J. Combin. Theory, Ser. B* 99, 1 (2009), 218–228.

[18] Martin Grohe, Thomas Schwentick, and Luc Segoufin. 2001. When is the evaluation of conjunctive queries tractable? In *Proceedings of the 33rd ACM Symposium on Theory of Computing (STOC'01)*. 657–666.

[19] Shlomo Hoory, Nathan Linial, and Avi Wigderson. 2006. Expander graphs and their applications. *Bull. Am. Math. Soc.* 43, 4 (2006), 439–561.

[20] Te C. Hu. 1969. *Integer Programming and Network Flows*. Technical Report. Department of Computer Sciences, University of Wisconsin—Madison.

[21] Russell Impagliazzo, Ramamohan Paturi, and Francis Zane. 1998. Which problems have strongly exponential complexity? In *Proceedings of the 39th IEEE Symposium on Foundations of Computer Science (FOCS'98)*. 653–662.

[22] Russell Impagliazzo, Ramomohan Paturi, and Francis Zane. 2001. Which problems have strongly exponential complexity?*J. Comput. Syst. Sci.* 63, 4 (2001), 512–530.

[23] Ken-ichi Kawarabayashi, Bojan Mohar, and Bruce Reed. 2008. A simpler linear time algorithm for embedding graphs into an arbitrary surface and the genus of graphs of bounded tree-width. In *Proceedings of the 49th IEEE Symposium on the Foundations of Computer Science (FOCS'08)*. IEEE, 771–780.

[24] Philip N. Klein and Dániel Marx. 2012. Solving Planar k-Terminal Cut in $O(n^{c\sqrt{k}})$ time. In *Proceedings of the International Colloquium on Automata, Languages, and Programming (ICALP'12)*. Springer, 569–580.

[25] Daniel Lokshtanov, Dániel Marx, Saket Saurabh, et al. 2013. Lower bounds based on the exponential time hypothesis. *Bull. EATCS* 3, 105 (2013).

[26] Daniel Lokshtanov, Saket Saurabh, and Magnus Wahlström. 2012. Subexponential parameterized odd cycle transversal on planar graphs. In *Proceedings of the IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS'12)*. 424–434.

[27] Adam W. Marcus, Daniel A. Spielman, and Nikhil Srivastava. 2015. Interlacing families I: Bipartite Ramanujan graphs of all degrees. *Ann. Math.* 182, 1 (2015), 307–325.

[28] Dániel Marx. 2007. On the optimality of planar and geometric approximation schemes. In *Proceedings of the 48th IEEE Symposium on Foundations of Computer Science (FOCS'07)*. 338–348.

[29] Dániel Marx. 2010. Can you beat treewidth? *Theory Comput.* 6, 1 (2010), 85–112.

[30] Dániel Marx. 2012. A tight lower bound for planar multiway cut with fixed number of terminals. In *Automata, Languages, and Programming*, Artur Czumaj, Kurt Mehlhorn, Andrew Pitts, and Roger Wattenhofer (Eds.). Springer, Berlin, 677–688.

[31] Dániel Marx. 2013. The square root phenomenon in planar graphs. In *Frontiers in Algorithmics and Algorithmic Aspects in Information and Management*, Vol. 7924. Springer.

[32] Dániel Marx and Michał Pilipczuk. 2014. Everything you always wanted to know about the parameterized complexity of Subgraph Isomorphism (but were afraid to ask). In *Proceedings of the 31st International Symposium on Theoretical Aspects of Computer Science (STACS'14)*, Vol. 25. 542–553.

[33] Dániel Marx and Michal Pilipczuk. 2015. Optimal parameterized algorithms for planar facility location problems using Voronoi diagrams. In *Proceedings of the 23rd European Symposium on Algorithms (ESA'15)*. 865–877.

[34] Dániel Marx, Marcin Pilipczuk, and Michał Pilipczuk. 2018. On subexponential parameterized algorithms for Steiner Tree and Directed Subset TSP on planar graphs. In *Proceedings of the 59th IEEE Symposium on Foundations of Computer Science (FOCS'18)*. 474–484.

[35] Dániel Marx and Anastasios Sidiropoulos. 2014. The limited blessing of low dimensionality: When $1 - 1/d$ is the best possible exponent for d-dimensional geometric problems. In *Proceedings of the 30th Symposium on Computational Geometry (SoCG'14)*. ACM, 67.

[36] Bojan Mohar. 1999. A linear time algorithm for embedding graphs in an arbitrary surface. *SIAM J. Discr. Math.* 12, 1 (1999), 6–26.

[37] Bojan Mohar and Carsten Thomassen. 2001. *Graphs on Surfaces*. Johns Hopkins University Press.

[38] Zoë Wood, Hugues Hoppe, Mathieu Desbrun, and Peter Schröder. 2004. Removing excess topology from isosurfaces. *ACM Trans. Graph.* 23, 2 (2004), 190–208.

Received October 2019; revised January 2021; accepted February 2021