HAL
open science

Contraction checking in graphs on surfaces

Marcin Kaminski, Dimitrios M. Thilikos

To cite this version:

Marcin Kaminski, Dimitrios M. Thilikos. Contraction checking in graphs on surfaces. STACS'12: 29th Symposium on Theoretical Aspects of Computer Science, Feb 2012, Paris, France. pp.182-193. hal-00678173

HAL Id: hal-00678173

https://hal.science/hal-00678173v1

Submitted on 3 Feb 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Contraction checking in graphs on surfaces

Marcin Kamiński*[1] and Dimitrios M. Thilikos†[2]

1 **Département d'Informatique**
Université Libre de Bruxelles
Marcin.Kaminski@ulb.ac.be

2 **Department of Mathematics**
National and Kapodistrian University of Athens
sedthilk@math.uoa.gr

Abstract

The Contraction Checking problem asks, given two graphs H and G as input, whether H can be obtained from G by a sequence of edge contractions. Contraction Checking remains NP-complete, even when H is fixed. We show that this is not the case when G is embeddable in a surface of fixed Euler genus. In particular, we give an algorithm that solves Contraction Checking in $f(h, g) \cdot |V(G)|^3$ steps, where h is the size of H and g is the Euler genus of the input graph G.

1998 ACM Subject Classification G.2.2 Graph Theory

Keywords and phrases Surfaces, Topological Minors, Contractions, Parameterized algorithms, Linkages

Digital Object Identifier 10.4230/LIPIcs.STACS.2012.182

1 Introduction

We consider simple finite graphs and use standard graph-theoretical terminology. For notions not define here, we refer the reader to Diestel [6] and to Mohar and Thomassen [17].

Contractions and topological minors. To *contract an edge* is to identify its two endpoints and remove the loop and multiple edges that have possibly been created. A graph H is a *contraction* of a graph G ($H <_c G$) if H can be obtained from G by a sequence of edge contractions. Deciding whether the input graph can be contracted to a fixed pattern is NP-complete, even for small pattern graphs – the smallest is an induced path on four vertices [3].

To *dissolve a vertex* of degree 2 is to contract one of the edges incident with it. A graph H is a *topological minor* of a graph G if H can be obtained from G by a sequence of vertex/edge deletions and vertex dissolutions. Recently, Grohe et al. proved that for every fixed graph H there exists an $\mathcal{O}(|V(G)|^3)$ time algorithm deciding whether H is a topological minor of G [13]. This is an FPT algorithm for this problem when parameterized by the size of H, that is, an algorithm with running time $g(|H|) \cdot |G|^{\mathcal{O}(1)}$. (For more information on parametrized complexity theory, see any of the books: Downey and Fellow [7], Flum and Grohe [10], or Niedermeier [20].)

Previous work on contractions. The problem of checking whether a graph is a contraction of another has attracted some attention. Perhaps the first systematic study of contractions

* Chargé de Recherches du FRS-FNRS.
† Supported by the project "Kapodistrias" (**ΑΠ** 02839/28.07.2008) of the National and Kapodistrian University of Athens (project code: 70/4/8757).

© M. Kamiński and D.M. Thilikos;
licensed under Creative Commons License NC-ND
29th Symposium on Theoretical Aspects of Computer Science (STACS'12).
Editors: Christoph Dürr, Thomas Wilke; pp. 182–193
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany
SYMPOSIUM ON THEORETICAL ASPECTS OF COMPUTER SCIENCE

was undertaken by Brouwer and Veldman [3]. According to the results of [3], checking if a graph is contractible to the induced cycle on four vertices or the induced path on four vertices is NP-complete. More generally, they prove that it is NP-complete for every bipartite graph with at least one connected component that is not a star. Looking at contractions to fixed pattern graphs is justified by the result by Matoušek and Thomas [15] who proved that deciding, given two input graphs G and H, whether G is contractible to H is NP-complete even when both G and H are trees.

Surface containment relations. Surface versions of contractions and topological minors can be defined for surface-embedded graphs. Formal definitions are presented in Section 2. For the purpose of this introduction, we only note that surface contractions and surface topological minors are surface-embedded versions of contractions and topological minors, respectively, that respect the embedding.

For every surface Σ and every pattern graph H, there exists a polynomial-time algorithm deciding whether a Σ-embedded graph can be contracted to H [14]. The algorithm is based on a combinatorial lemma that allows to reduce the problem of testing for contraction in a surface-embedded graph to a constant number of tests for surface topological minors in its dual. The procedure is polynomial for every fixed graph H; however, the degree of the polynomial depends on the size of H. Is it possible to design an FPT algorithm for this problem when parameterized by the size of H?

The main obstacle is testing for surface topological minors. If there existed an FPT algorithm for deciding if a surface-embedded input graph contains a pattern graph H as a surface topological minor, then the machinery of [14] would imply an FPT algorithm for contraction checking. Surface topological minors are different from topological minors as they are defined for surface-embedded graphs and respect the embedding. While it is possible to reduce topological minor testing to surface topological minor testing, the latter is not known to be FPT-reducible to the former.

In this paper we overcome these difficulties and show that testing whether a surface-embedded graph is contractible to a given pattern is FPT, when parameterized by the size of the pattern.

The irrelevant vertex technique. A core technique from Graph Minors by Robertson and Seymour that has been especially prolific in algorithmic research is the following win/win approach. If the treewitdh of the input graph is small (less than a certain constant c depending on the problem parameter), apply dynamic programming and solve the problem in FPT time with respect to c; otherwise, exploit the existence of a subdivision of a large wall in the input graph (its size depends on c). In the latter case, one can usually find an *irrelevant vertex* – a vertex that can be safely removed from the graph without changing the solution. Then, the algorithm is recursively applied to the new graph so that, eventually, the treewidth of the graph drops below c to make the dynamic programming approach applicable.

Our approach. We follow this general scheme, however, we additionally prove that one can assume that the subgraph containing a large subdivided wall is of bounded treewidth. More precisely, for every positive integer h and a surface Σ, there exist constants t and T such that in every Σ-embedded graph of treewidth at least t there exists a disk in Σ such that the graph induced by the vertices inside the disk is of treewidth at most T and contains a subdivision of a wall of height h. This assumption comes in handy in our proof. We also believe that this lemma is of independent interest and can be applied to other problems.

Having found a subgraph of bounded treewidth containing a large subdivided wall, we consider a collection of nested cycles from the wall. For each cycle, we check what subpatterns of the guest graph can be seen as surface topological minors of its interior with a

Relation	planar graph	graphs on surfaces	all graphs
(induced) subgraph	FPT [8]		W[1]-hard
minor	FPT[22]		
topological minor	FPT [13]		
weak/strong immersion	FPT [13]		
induced minor	FPT [9]	OPEN	para-NP-complete [9]
contraction	FPT [this paper]		para-NP-complete [3]

Table 1 Overview of parameterized complexity status of containment relations in graphs.

"certain attachment" to the boundary of the cycle. This attachment determines the possible ways such a pattern should be extended outside the cycle towards matching the structure of the host graph. This is encoded as a *characteristic* function of each cycle. A key property is that the characteristic function is *monotone* – whatever can be attached to a cycle, can also be attached to subsequent cycles in the collection.

The main idea is to determine a collection of consecutive cycles with the same characteristic function, which is now feasible since this computation takes place in a graph of bounded treewidth. If this collection is "sufficiently large" then the monotonicity property implies that every sub-pattern of the guest graph can be also located away from some "safe" cycle and this is proved by making use of the Unique Linkage Theorem of Robertson and Seymour from [21, 23]. Then the safe cycle contains an irrelevant vertex that is removed and the procedure recurses until the host graph has bounded treewidth.

Table 1 summarizes the current state of research on parameterized complexity of containment relations, including the contribution of this paper.

In this extended abstract we give a detailed outline of the algorithm. The complete presentation of the algorithm and the proof will appear in the journal version of this paper.

2 Definitions

Surfaces. A *surface* Σ is a compact 2-manifold without boundary (we always consider connected surfaces). Whenever we refer to a Σ-*embedded graph* G we consider G accompanied by some embedding of it in Σ without crossings. To simplify notation, we do not distinguish between a vertex of G and the point of Σ used in the drawing to represent the vertex or between an edge and the line representing it. Given an edge e, we denote by $\overline{e}$ the set of its endpoints (clearly, $1 \leq |\overline{e}| \leq 2$). We also consider a graph G embedded in Σ as the union of the points corresponding to its vertices and edges. That way, a subgraph H of G can be seen as a graph H, where $H \subseteq G$. We refer to the book of Mohar and Thomassen [19] for more details on graph embeddings. The *Euler genus* of a graph G is the minimum integer γ such that G can be embedded on a surface of the Euler genus γ.

Given a Σ-embedded graph G, we denote by $F(G)$ the set of its faces, i.e. the set of connected components of the set $\Sigma \setminus G$. We say that a face in $F(G)$ is *trivial* if it is incident with at most two edges. An edge is *trivial* if it is incident with a trivial face. A loop of G is an edge with one endpoint. We say that a loop e is *singular* if it is either non-contractible or it is contractible and both connected components of $\Sigma \setminus e$ contain vertices of G.

The *surface contraction of an edge* e in a Σ-embedded graph G is the graph $G' = G\backslash_{\Sigma} e$ defined as follows. In case e is non-singular, G' is the graph obtained if we identify the closure of all points of e to a single vertex. In case e is singular the G' is the graph obtained

from G after removing all points of e. Notice that surface contractions are defined in a way that preserves surface integrity.

Let H and G be two Σ-embedded graphs. We say that H is a *surface contraction* of G, denoted by $H \leq_{\mathrm{c}}^{\Sigma} G$, if H can be obtained from G by a (possibly empty) sequence of operations that may be either surface contractions of edges or removals of trivial edges. Finally, we say that H is a *surface minor* of G, if H is a surface contraction of some subgraph of G.

Isomorphism. Let A_1 and A_2 be graphs and let $\psi\colon V(A_1) \to V(A_2)$ be a bijection. We say that A_1 and A_2 are *ψ-isomorphic* if for each pair $x, y \in V(A_1)$ it holds that $\{x, y\} \in E(A_1)$ if and only if $\{\psi(x), \phi(y)\} \in E(A_2)$. The *edge extension* of ψ, denoted by $\psi^e : V(A_1) \cup E(A_1) \to V(A_2) \cup E(A_2)$ extends ψ so to incorporate the correspondence between the edges of A_1 and the edges of A_2 implied by ψ.

Topological isomorphism. Let A_i be Σ_i-embedded graphs $i \in \{1, 2\}$. Suppose also that Σ_1 is homeomorphic to Σ_2. Let $\psi : V(A_1) \to V(A_2)$ be a bijection from $V(A_1)$ to $V(A_2)$. We say that A_1 is ψ-topologically isomorphic to A_2 is there is a homeomorphism $\phi : \Sigma_1 \to \Sigma_2$ such that ψ is an isomorphism from A_1 to A_2 and ψ^e is induced by the restriction of ϕ in $V(A_1)$. Notice that the bijection ψ above is an isomorphism between A_1 and A_2.

Surface topological minor. Let Σ be a surface and $\mathcal{G}$ be a Σ-embedded graph. Given a set $\mathcal{P}$ of internally disjoint extended paths of G, we define $G_{\mathcal{P}}$ as the Σ-embedded graph created if we first remove from G each edge not in a path in $\mathcal{P}$ and then replace each extended path (P, A) in $\mathcal{P}(G)$ by the extended path $((\overline{e}, \{e\}), A)$ where e is a new edge and $\overline{e} = A$.

Let Σ be a surface and (G, S_G) and (H, S_H) be two rooted Σ-embedded graphs. Let also σ be a bijection from S_G to S_H. We say that (H, S_H) is a *surface σ-rooted topological minor* of (G, S_G), and we denote it by $(H, S_H) \leq_{\sigma}^{\Sigma} (G, S_G)$ if there is a collection $\mathcal{P}$ of internally disjoint extended paths in G such that $G_{\mathcal{P}}$ is ψ-topologically isomorphic to H for some bijection $\psi : V(G_{\mathcal{P}}) \to V(H)$ where $\sigma \subseteq \psi$. When $S_G = S_H = \emptyset$, we say that H is a *surface topological minor* for G and denote it by $H \leq_{\mathrm{stm}}^{\Sigma} G$.

The main technical result of [14] is an equivalence between surface contractions in a surface-embedded graph and surface topological minors in its dual. A multigraph is called *thin* if it has no two parallel edges bounding a 2-face. (In particular, simple graphs are thin.) For a surface Σ and a simple Σ-embedded graph H, let $C_{\Sigma}(H)$ be a maximal set of thin Σ-embedded multigraphs that have the same adjacencies between their vertices as H (that is, forgetting multiple edges) such that they are all pairwise not topologically isomorphic. The set $C_{\Sigma}(H)$ is finite (Lemma 5 in [14]).

▶ Proposition 1 ([14]). Let G and H be graphs. Suppose also that G is embedded in a surface Σ and let G^* be its dual. Then $H \leq_{\mathrm{c}}^{\Sigma} G$ if and only if there exists a graph $\hat{H} \in C_{\Sigma}(H)$ such that $\hat{H}^* \leq_{\mathrm{stm}}^{\Sigma} G^*$.

3 Description of the algorithm

Let G and H be the host and the guest graph respectively. We denote by n the number of vertices in G. Also, in order to maintain only one parameter during the description of the algorithm, we assume that $h = |E(H)| + |V(H)| + \mathbf{eg}(G)$, where $\mathbf{eg}(G)$ is the Euler genus of G. For simplicity, we will use the notation $O_h(n^{\alpha})$ instead of $f(h) \cdot n^{\alpha}$ where f is some computable function of h.

General framework. Following the idea of the irrelevant vertex technique, introduced by Robertson and Seymour in [22], our first step is to check whether the treewidth of G is at

most $f_0(H) + h + 1$ where $f_0 : \mathbb{N} \to \mathbb{N}$ is a suitable function of H. This can be done in $O_h(n)$ steps because of the results in [2]. If $\mathbf{tw}(G) < f_0(h) + h + 1$, then the problem can be solved by the dynamic programming algorithm of [1] in $O_h(n)$ steps (this also follows from Courcelle's theorem [4] and the fact that contraction checking is expressible in Monadic Second Order Logic). So we may assume that $\mathbf{tw}(G) \geq f_0(h) + h + 1$. Also using the algorithm in [18] we may consider that G is optimally 2-cell embedded in some surface Σ of Euler genus $\mathbf{eg}(G)$. Let G^* be the dual embedding of G in Σ. From [16], the treewidth of a Σ 2-cell embedded graph and the treewidth of its dual cannot differ more than $\mathbf{eg}(\Sigma) + 1$. Therefore $\mathbf{tw}(G^*) \geq f_0(h)$. From Proposition 1, H is a contraction of G if and only if for some Σ-embedded graph in $\hat{H} \in C_\Sigma(H)$ it holds that $\hat{H}^* \leq^{\Sigma}_{\text{stm}} G^*$. Recall that the size of each graph in $C_\Sigma(H)$ depends only on H and $\mathbf{eg}(G)$ and therefore is bounded by $f_1(h)$ for some function f_1.

Our goal is to give an $O_h(n^2)$ step procedure with the following specifications:

Procedure Irrelevant Edge Detection(G, Σ)
Input: a graph G' of treewidth at least $f_0(h)$ that is 2-cell embedded in a surface Σ of Euler genus $\leq h$.
Output: an edge $e' \in E(G')$ such that $G' \setminus e$ remains 2-cell embedded in Σ and for every Σ-embedded graph H' of size at most $f_1(h)$, it holds that

$$H' \leq^{\Sigma}_{\text{stm}} G' \Leftrightarrow H' \leq^{\Sigma}_{\text{stm}} G' \setminus e'.$$

Actually, function f_0 should be chosen to be "sufficiently big" so it is possible to find an irrelevant edge.

Let e^* be the output of Irrelevant Edge Detection(G^*, Σ). Using the proof of Proposition 1, we may find an edge $e^* \in E(G^*)$ such that if e^* it is the dual edge of $e \in E(G)$, then H is a contraction of G if and only if H is a contraction of G/e. That way we reduce, in $O_h(n^2)$ steps, the problem of checking whether $H \leq_c G$ to the problem whether $H \leq_c G_{\text{new}} = G/e$. Clearly, we may again check whether $\mathbf{tw}(G_{\text{new}}) < f_0(h) + h + 1$ and either solve the problem by dynamic programming or again apply the Irrelevant Edge Detection procedure on G_{new}. Since the new graph is always smaller than the previous, applying the same steps, the algorithm will stop and produce a correct solution. As this will occur in less than n repetitions, the whole algorithm will take $O_h(n^3)$ steps, as claimed.

Given the above framework, what remains is to describe how the Irrelevant Edge Detection procedure works.

Big walls of small treewidth. It follows from the results in [5, 11, 12] that every Σ-embeddable graph of big enough treewidth contains as a subgraph a subdivision of a wall of given height and width (where height and width are defined in the obvious way). Also, by the same results, we can assume that this subdivision is "flat in the surface" in the sense that its perimeter is a contractible cycle of the embedding (i.e. handles are outside the wall). An example of such a subdivided wall is depicted in Figure 1 (for simplicity, we do not depict the subdivision vertices). We need the following Lemma:

▶ **Lemma 1.** *There are functions t_1 and t_2 such that, for every κ, every graph G that is embedded in a surface Σ of Euler genus g and has treewidth at least $t_1(\kappa, g)$, contains a subgraph R such that*

- *R is the subdivision of a wall of height and width equal to k,*
- *R is drawn inside a closed disk Δ bounded by its perimeter, and*

- *$\Delta \cap G$, i.e. the part of the graph that lies inside the perimeter of R, has treewidth upper bounded by $t_2(\kappa, g)$.*

Also, such a graph R can be computed in $O_h(n^2)$ steps.

Proof. The following claim can easily be derived by Lemma 4 in [11].

Claim. Let G be a graph embedded in a surface Σ of Euler genus g and let i be a positive integer. If $\mathbf{tw}(G) \geq 48i(g+1)$, then G contains a subdivided wall R of height i and width i as a subgraph and R is drawn inside a closed disk Δ of Σ bounded by the perimeter of G'.

Let $t_1(\kappa, g) = 48\kappa(g+1)$ and $t_2(\kappa, g) = 48(\kappa+1)(g+1)$. Apply the following routine on G.

1. Let $G' := G$.
2. While $\mathbf{tw}(G') \geq t_2(\kappa, g)$ do
3. let $i = \kappa + 2$,
4. let R' be a subdivided wall of height i, as in the above claim, and
5. update G' to the subgraph of G' induced by the vertices in the *strict* interior of the perimeter of R'.
6. Output G'.

Notice that the output of the above routine has always treewidth at most $t_2(\kappa, g)$. If the above algorithm never enters the loop of lines 3–5, then $\mathbf{tw}(G') = \mathbf{tw}(G) \geq t_1(\kappa, g)$ and, because of the above claim for $i = k$, G contains the desired subdivided wall R of height k. If this is not the case, then because of the stripping of Line 5, G' (and thus G as well) contains a wall R of height $i - 2 = k$, as required. ◀

The third assertion of Lemma 1 is important for our algorithm, as it implies that all subgraphs of G that are inside the outer cycle have bounded treewith and therefore, for these graphs, it is possible to answer queries on (rooted) surface topological minor containment in $O_h(n)$ steps.

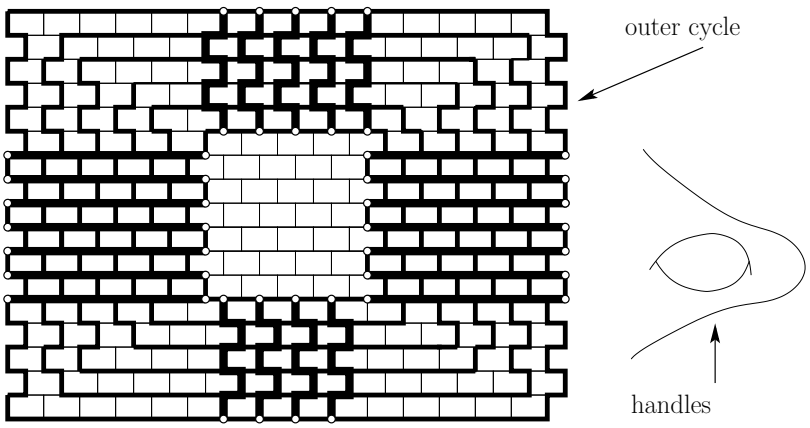

Figure 1 A wall of height 17 and width 15 together with a railed annulus of 6 cycles and 23 rails in it.

Cycles, rails, and tracks. Notice now that inside the perimeter of a subdivided wall of "big" enough height and width, one may distinguish a collection of nested cycles $\mathcal{A} = \{C_1, \ldots, C_r\}$

all met by a collection of paths $\mathcal{W} = \{W_1, \ldots, W_q\}$ (we call them *rails*) in a way that the intersection of a rail and a cycle is always a path. We can also assume that, among these cycles, C_r is the perimeter of the subdivided wall and we call it the *outer cycle*.

See Figure 1 for an example of how to extract 6 cycles and 23 rails from a (subdivided) wall of height 17 and width 15. We call this pair $(\mathcal{A}, \mathcal{W})$ of collections of cycles and rails *railed annulus* and observe that all rails and cycles are contained inside the outer cycle. Moreover, given that we need k_1 cycles and k_2 rails, we can always find them in a subdivided wall of big enough height and width. Combining this fact with Lemma 1, we derive the following.

▶ **Lemma 2.** *There exist functions t_3 and t_4, such that every graph G that is embedded in a surface Σ of Euler genus g and has treewidth at least $t_3(r, q)$ contains a* railed annulus *$(\mathcal{A}, \mathcal{W})$ if r cycles and q rails such that every subgraph of G that is entirely inside the outer cycle of $\mathcal{A}$ has treewidth at most $t_4(r, q)$.*

For a more abstract visualization of a railed annulus with 9 cycles and 24 rails, see Figure 2.

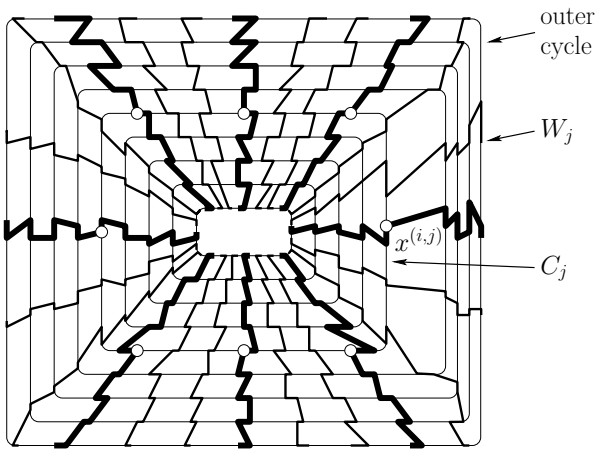

Figure 2 A railed annulus of 9 cycles and 24 rails. Among them, we distinguish 8 tracks.

For the purposes of our algorithm, we distinguish some proper subset of the rails and we call them *tracks*. For each cycle C_i of a railed annulus and for each rail W_h, we denote by $x^{(i,j)}$ the last vertex, starting from inside, of W_h that is a vertex of C_i. For the i-th cycle (counting from inside to outside) we denote by $X^{(i)}$ the set of all $x^{(i,j)}$'s on it (in Figure 2, $X^{(5)}$ consists of the white vertices). Also, for each i, we denote by $\Delta^{(i)}$ the inner closed disk bounded by C_i and by $G^{(i)}$ the subgraph of G that is is inside $\Delta^{(i)}$.

Crossings of a pattern graph. Let H be a Σ-embedded pattern graph of at most h edges and let $\tilde{\Delta}$ be a closed disk of Σ. The notion of a graph J that is $\tilde{\Delta}$-excised by H is visualized in Figure 5. Notice that J is embedded inside $\tilde{\Delta}$ and contains new vertices (the white vertices, denoted by X) that are the points of intersection of H with the boundary of $\tilde{\Delta}$. The number of these white vertices is the *crossing number* of J. We see each $\tilde{\Delta}$-excised graph J as being embedded inside the disk Δ. We also consider its *enhancement* $J_{\tilde{\Delta},X}$ by adding edges between boundary vertices as depicted in Figure 5. We say tha two $\tilde{\Delta}$-excised graphs J^1 and J^2 are *equivalent* if their enhancements $J^1_{\Delta,\mathrm{X}}$ and $J^2_{\Delta,\mathrm{X}}$ are topologically isomorphic.

We also define the same enhancement for each graph $G^{(i)}$ and we denote it by $G^{(i)}_{\Delta^{(i)},X^{(i)}}$ (see the left part of Figure 5).

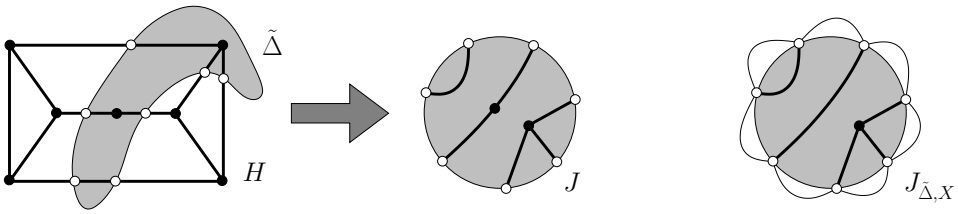

Figure 3 A graph J that is $\tilde{\Delta}$-excised by H and its enhanced version $J_{\tilde{\Delta},X}$ (X consists of the white vertices).

Attached topological minors. We set up a repository $\mathbf{H}_h$ of all graphs J that can be $\tilde{\Delta}$-excised by H with crossing number $f_4(h)$ where f_4 is a function to be determined later. Clearly, the size of $\mathbf{H}_h$ depends exclusively on h. Our next step is to set up a 0/1-vector χ_i that encodes, for every $J \in \mathbf{H}_h$ and every mapping $\rho : X \to X^{(i)}$, whether $J_{\tilde{\Delta},X}$ is a surface topological minor of $G^{(i)}_{\Delta^{(i)},X^{(i)}}$, where the vertices of X are mapped to vertices of $X^{(i)}$ as indicated by ρ. When this happens, we say that J is a *ρ-attached topological minor of* $G^{(i)}$). For an example of such a mapping, see the right part of Figure 5.

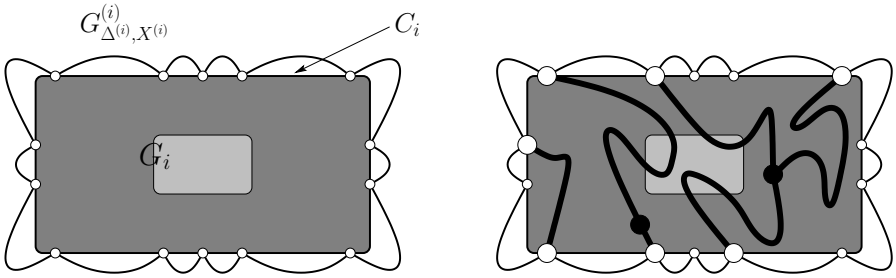

Figure 4 The graph $G^{(i)}_{\Delta^{(i)},X^{(i)}}$ and a realization of J as a ρ-attached topological minor of $G^{(i)}$.

Detecting an irrelevant edge. As each $G^{(i)}$ has bounded treewidth and the property of being a ρ-attached topological minor can be expressed in MSOL, χ_i can be computed in $O_h(n)$ steps and can be encoded in space that depends exclusively on h. It is important to notice that the vector sequence $\chi_1, \ldots, \chi_r$ is *monotone* in the sense that if a graph J is a ρ-attached topological minor of G^i, then it is also a ρ-attached topological minor of $G^{i'}$ for $i' > i$. By a pigeonhole argument, if the number of the cycles in the railed annulus is big enough, then there should exist a sub-collection $C_{\theta+1}, \ldots, C_{\theta+l}$ of consecutive cycles where $\chi_{\theta+1} = \ldots = \chi_{\theta+l}$, i.e., where the members of $\mathbf{H}_r$ behave the same as ρ-attached topological minors in their interiors (here l will be chosen to be as big as required for the correctness of our proofs). We call the seqence $C_{\theta+1}, \ldots, C_{\theta+l}$ *frozen* and observe that it can be detected algorithmically in $O_h(n)$ steps. In other words, we have the following:

▶ **Lemma 3.** *There exists some function $g : \mathbb{N} \to \mathbb{N}$ such that for every two positive integers h and l, every Σ-embedded graph G with a (r, q)-railed annulus $(\mathcal{A}, \mathcal{W})$ where $r \geq g(h) \cdot l$, and every $I \subset \{1, \ldots, q\}$ there is an integer $\theta \in \{0, \ldots, r-l\}$, such that the sequence $\{\chi_1, \ldots, \chi_r\}$ contains a subsequence $\{\chi_{\theta+1}, \ldots, \chi_{\theta+l}\}$ of l consecutive equal vectors. Moreover, there is an algorithm that, given h, l, G, $(\mathcal{A}, \mathcal{W})$, and I, outputs θ in $\phi(h, \mathbf{tw}(G^{(r)})) \cdot n$ steps, for some function ϕ.*

We claim that any edge in a non-track rail that lies between C_r and C_{r+1} is an irrelevant edge. In other words, the procedure Procedure Irrelevant Edge Detection(G, Σ) is the following:

Procedure Irrelevant Edge Detection(G, Σ)
1. Compute $\mathbf{H}_h$.
2. Find, using Lemma 2, a railed annulus $(\mathcal{A}, \mathcal{W})$
   in G with $r = g(h) \cdot t_3(h)$ cycles and $t_4(h)$ rails.
3. Pick a proper subset I of $\{1, \ldots, q\}$ of size $t_5(h)$
   and call the rails in $\{W_i \mid i \in I\}$ *tracks*.
4. Apply Lemma 3, using $(\mathcal{A}, \mathcal{W})$ and its tracks, in order to
   detect a frozen sequence $C_{\theta+1}, \ldots, C_{\theta+l}$ in $\mathcal{A}$.
6. Let $i \in \{1, \ldots .r\} \setminus I$ and let e be an edge of W_i that lies
   between $C_{\theta+1}$ and $C_{\theta+2}$, i.e. an edge in $W_i \cap (\Delta_{\theta+2} \setminus C_{\theta+1} \setminus \Delta_{\theta+2})$.
7. Output e.

The functions t_3, t_4, and t_5 above, depend on H and the genus of G and will be determined later so that the algorithm is correct.

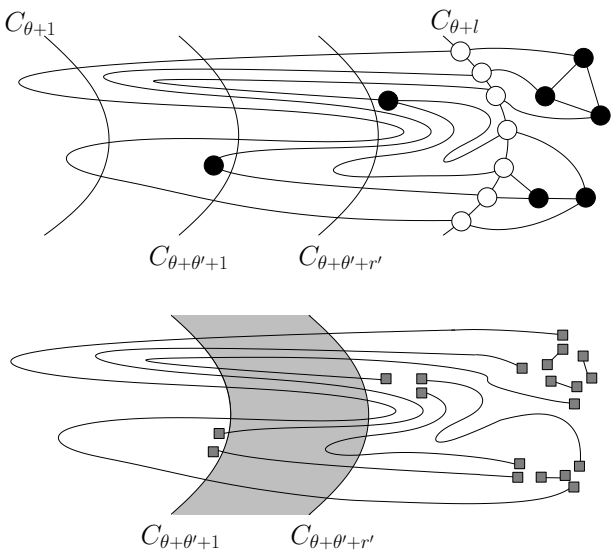

Figure 5 The upper figure depicts a realization of H as a topological minor of G. The annulus defined by the cycles $C_{\theta+\theta'+1}$ and $C_{\theta+\theta'+r'}$ does not contain any image of a vertex in H. The lower figure shows the corresponding linkage.

4 Correctness of the algorithm

This section contains a sketch of the proof that irrelevant edges are indeed irrelevant.

Linkage extraction. Suppose that H is a surface topological minor of G. Our purpose is to find a realization of H as a surface topological minor of G in a way that avoids the irrelevant edge. For this we fix our attention in the "frozen" annulus defined by the cycles $C_{\theta+1}$ and $C_{\theta+l}$. As H has at most $2 \cdot h$ vertices, there should be a big enough sub-annulus that does not contain any images of the vertices of H. Assume that this sub-annulus contains the

r' cycles $C_{\theta+\theta'+1}, \ldots, C_{\theta+\theta'+r'}$. Notice that H defines a collection of disjoint paths whose terminals are outside this annulus. This collection is a h'-*linkage* (i.e. a subgraph consisting of a collection of at most h' disjoint paths) for some $h' \leq h$ and we denote it by $\mathcal{L}'$ (see Figure 5).

Linkage replacement. The terminals of a linkage are the endpoints of its paths. Recall that the terminals of the linkage $\mathcal{L}$ that we detected in the previous paragraph has all its linkages outside the closed annulus defined by the cycles C_1 and C_r. We call such a linkage $\mathcal{A}$-avoiding linkage. Our next step is to prove the following lemma:

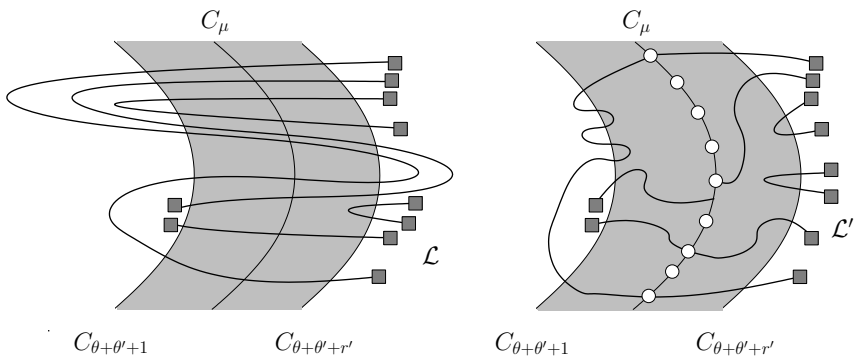

Figure 6 The replacement of linkage $\mathcal{L}$ by a linkage $\mathcal{L}'$. (We do not depict paths that are entirely outside the sub-annulus. Also, for reasons of simplicity we represent the intersection of all, except from one, paths with C_μ by a single vertex instead of a path.)

▶ **Lemma 4.** *There exist functions t_3, t_4, and t_5 such that the following hold: If h is a positive integer h, G a Sigma-embedded graph with a railed annulus $(\mathcal{A}, \mathcal{W})$ with $r = t_3(h)$ cycles and $q = t_4(h)$ rails, $\mathcal{L}$ an $\mathcal{A}$-avoiding linkage $\mathcal{L}$ and subset I a proper subset of $\{1, \ldots, q\}$ where $|I| = t_5(h)$, then there is an $\mathcal{A}$-avoiding linkage $\mathcal{L}$ with the following properties:*

- *the paths of $\mathcal{L}$ link the same terminals as the paths in $\mathcal{L}'$,*
- *no more than $t_5(h)$ paths in $\mathcal{L}'$ cross the "middle" cycle $C_{\lceil r/2 \rceil}$ and, when this happens, their intersection will be just a path,*
- *when we orient such a path from inside to outside, its last in C_μ should always be a vertex of $X^{(\mu)}$.*

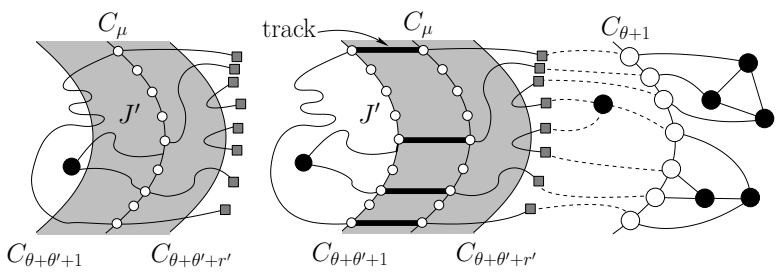

Figure 7 Two different realizations of J' as ρ-attached topological minors of G^μ. The one on the right avoids the irrelevant edge.

The proof of the above lemma is quite technical and uses the "vital linkage" Theorem of

Roberstong and Seymour in [23] (actually the function t_5 is directly taken from [23]). An example of this linkage replacement is depicted in Figure 6.

Pattern displacement. Our next step is to observe that the new linkage gives rise to a graph J' of $\mathbf{H}_h$ that is a ρ-attached topological minor of $G^{(\mu)}$. Recall that $\chi_{\theta+\theta'+1} = \chi_\mu$. Therefore, J' is also ρ'-attached topological minor of $G^{(\theta+\theta'+1)}$ where ρ' is the "left-side displacement" of ρ from C_μ to $C_{\theta+\theta'+1}$. But then, we may use the segments of the tracks that are cropped by the annulus defined by C_μ and $C_{\theta+\theta'+1}$ to realize J' as a ρ'-attached topological minor of $G^{(\mu)}$ in a way that rails that are not tracks are avoided (see Figure 7).

Clearly, the new realization of J' avoids the irrelevant edge and can be extended to a realization of H as a surface topological minor of G (see the right part of Figure 7). This means that the irrelevant edge is indeed irrelevant and this yields the correctness of procedure Irrelevant Edge Detection(G, Σ).

5 Open problem

We prove that contraction checking is FPT for graphs on surfaces. To complete Table 1 it would be interesting to know the parametrized complexity of induced minor checking for graphs on surfaces.

Acknowledgments

We wish to thank the anonymous reviewers of previous versions of this paper. Their comments and remarks were helpful in improving the presentation.

References

1 Isolde Adler, Frederic Dorn, Fedor V. Fomin, Ignasi Sau, and Dimitrios M. Thilikos. Fast minor testing in planar graphs. In *Algorithms - ESA 2010, 18th Annual European Symposium (1)*, volume 6346 of *Lecture Notes in Computer Science*, pages 97–109. Springer, 2010.

2 Hans L. Bodlaender. A linear-time algorithm for finding tree-decompositions of small treewidth. *SIAM J. Comput.*, 25(6):1305–1317, 1996.

3 A. E. Brouwer and H. J. Veldman. Contractibility and NP-completeness. *Journal of Graph Theory*, 11(1):71–79, 1987.

4 Bruno Courcelle. The monadic second-order logic of graphs. I. Recognizable sets of finite graphs. *Inf. Comput.*, 85(1):12–75, 1990.

5 Erik D. Demaine, Fedor V. Fomin, Mohammadtaghi Hajiaghayi, and Dimitrios M. Thilikos. Subexponential parameterized algorithms on bounded-genus graphs and H-minor-free graphs. *J. Assoc. Comput. Mach.*, 52(6):866–893, 2005.

6 Reinhard Diestel. *Graph Theory.* Springer-Verlag, Electronic Edition, 2005.

7 R.G. Downey and M. R. Fellows. *Parameterized Complexity.* Springer-Verlag, 1999.

8 David Eppstein. Subgraph isomorphism in planar graphs and related problems. *J. Graph Algorithms Appl.*, 3(3), 1999.

9 Michael R. Fellows, Jan Kratochvíl, Martin Middendorf, and Frank Pfeiffer. The complexity of induced minors and related problems. *Algorithmica*, 13(3):266–282, 1995.

10 J. Flum and M. Grohe. *Parameterized complexity theory.* Texts in Theoretical Computer Science. An EATCS Series. Springer-Verlag, Berlin, 2006.

11 Fedor V. Fomin, Petr A. Golovach, and Dimitrios M. Thilikos. Contraction obstructions for treewidth. *J. Comb. Theory, Ser. B*, 101(5):302–314, 2011.

12 J. F. Geelen, R. B. Richter, and G. Salazar. Embedding grids in surfaces. *European J. Combin.*, 25(6):785–792, 2004.

13 Martin Grohe, Ken-ichi Kawarabayashi, Dániel Marx, and Paul Wollan. Finding topological subgraphs is fixed-parameter tractable. *To appear in Proceedings of the 43rd ACM Symposium on Theory of Computing(STOC'11)*, 2011.

14 Marcin Kamiński, Daniël Paulusma, and Dimitrios M. Thilikos. Contractions of planar graphs in polynomial time. In Mark de Berg and Ulrich Meyer, editors, *ESA*, volume 6346 of *Lecture Notes in Computer Science*, pages 122–133. Springer, 2010.

15 Jirí Matousek and Robin Thomas. On the complexity of finding iso- and other morphisms for partial k-trees. *Discrete Mathematics*, 108(1-3):343–364, 1992.

16 Frédéric Mazoit. Tree-width of hypergraphs and surface duality. *CoRR*, abs/1006.3167, 2010.

17 B. Mohar and C. Thomassen. *Graphs on Surfaces*. The Johns Hopkins University Press, 2001.

18 Bojan Mohar. A linear time algorithm for embedding graphs in an arbitrary surface. *SIAM J. Discrete Math.*, 12(1):6–26, 1999.

19 Bojan Mohar and Carsten Thomassen. *Graphs on surfaces*. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, Baltimore, MD, 2001.

20 Rolf Niedermeier. *Invitation to fixed-parameter algorithms*, volume 31 of *Oxford Lecture Series in Mathematics and its Applications*. Oxford University Press, Oxford, 2006.

21 Neil Robertson and Paul D. Seymour. Graph Minors. XXII. Irrelevant vertices in linkage problems. *Journal of Combinatorial Theory, Series B*. (to appear).

22 Neil Robertson and Paul D. Seymour. Graph Minors XIII. The disjoint paths problem. *J. Comb. Theory, Ser. B*, 63(1):65–110, 1995.

23 Neil Robertson and Paul D. Seymour. Graph Minors. XXI. Graphs with unique linkages. *Journal of Combinatorial Theory, Series B*, 99(3):583–616, 2009.