Treewidth, crushing and hyperbolic volume

CLÉMENT MARIA

JESSICA S PURCELL

The treewidth of a 3–manifold triangulation plays an important role in algorithmic 3–manifold theory, and so it is useful to find bounds on the treewidth in terms of other properties of the manifold. We prove that there exists a universal constant c such that any closed hyperbolic 3–manifold admits a triangulation of treewidth at most the product of c and the volume. The converse is not true: we show there exists a sequence of hyperbolic 3–manifolds of bounded treewidth but volume approaching infinity. Along the way, we prove that crushing a normal surface in a triangulation does not increase the carving-width, and hence crushing any number of normal surfaces in a triangulation affects treewidth by at most a constant multiple.

1 Introduction

This paper concerns invariants of 3–manifolds that are of interest both in geometry and in computational topology. For computational purposes, 3–manifolds are often expressed by a *triangulation*, that is by gluing a collection of tetrahedra. For example, this is true for 3–manifold software `SnapPea`, developed by Weeks in the early 1980s, now maintained and distributed as `SnapPy` [14], and for `Regina`, developed by Burton [8]. These programs have been influential in the development of 3–manifold geometry and topology. Computational topology considers the running time of algorithms. For an algorithm that takes a triangulation as input, the running time frequently depends on some measure of the "simplicity" of the triangulation. Note, however, that a 3–manifold can have many different triangulations. Therefore, it is important to produce triangulations that are as simple as possible. Here, we will evaluate the simplicity of a triangulation by its *treewidth*.

The treewidth of a triangulation is a measure of the sparsity of the gluing relations between tetrahedra; see Definition 2.7. It was first developed in graph theory by Robertson and Seymour [31], then adapted to 3–manifold triangulations. In recent years, several algorithms have been developed that are highly efficient for triangulations

with low treewidth — see Burton, Maria and Spreer [11; 12] — and so we would like to find triangulations of 3–manifolds with treewidth bounded in terms of well-understood properties of the manifold.

One property is geometry. By the geometrisation theorem proved by Perelman [28; 29] (or see Kleiner and Lott [24]) every closed orientable 3–manifold decomposes into geometric pieces, and the hyperbolic pieces are among the most prevalent and least understood. If a closed 3–manifold admits a hyperbolic structure, then that structure is a topological invariant of the manifold — see Mostow [27] — and so it is natural to ask if the hyperbolic geometric properties of the manifold can bound treewidth of a triangulation.

A hyperbolic invariant that has received much attention is the hyperbolic volume. By work of Jørgensen and Thurston, a hyperbolic 3–manifold M that has a lower bound on injectivity radius admits a triangulation with $O(\mathrm{vol}(M))$ tetrahedra — see Thurston [36], and also Kobayashi and Rieck [25]. However, if we put no restrictions on injectivity radius, then no such result holds: for a sufficiently large constant $C > 0$, there are infinitely many closed hyperbolic 3–manifolds with volume bounded above by C, and therefore a finite number of tetrahedra cannot triangulate them all. For example, such manifolds are obtained by Dehn filling a hyperbolic manifold with finite volume, using the fact that volume decreases under Dehn filling [36]; or see Benedetti and Petronio [4, Chapter E]. Nevertheless, we prove in this paper that any hyperbolic 3–manifold with bounded volume admits a triangulation with bounded treewidth.

Theorem 1.1 *There exists a universal constant $c > 0$ such that a hyperbolic 3–manifold M with volume $\mathrm{vol}(M)$ admits a triangulation with treewidth at most $c \cdot \mathrm{vol}(M)$.*

In computer science, parametrised complexity classifies computational difficulty in terms of multiple parameters as input. Some problems that are known to require super-polynomial time in terms of the input alone, under standard computational complexity assumptions, can be solved by algorithms that are exponential in one fixed parameter, but only polynomial in the size of another. Thus they can be solved efficiently for low values of the first parameter. An important example of this from graph theory is Courcelle's theorem, which states that many graph theory problems can be decided in linear time in the treewidth of the graph [13]. Recently, Courcelle's theorem has been adapted to 3–manifold topology by Burton and Downey [10]. Along with the

rich theory of parametrised algorithms and standard dynamic programming techniques, this has led to the development of several algorithms in 3–manifold topology that are both theoretically and practically efficient provided the input triangulation has small treewidth. Some of these parametrised algorithms have been implemented in the 3–manifold software `Regina` [8], and have led to significant improvement in practical computations.

In practice, the treewidth parameter is strongly dependent on the triangulation chosen for representing a manifold, and obtaining low treewidth triangulations can be difficult; see Maria and Spreer [26] for a discussion. Unfortunately, a manifold that has a simple topological or geometric description can often be represented by a triangulation that has extremely large treewidth, with no obvious combinatorial simplifications. Therefore it is important to identify triangulations of a manifold whose treewidth is bounded by topological or geometric properties of the manifold, as in Theorem 1.1.

We also consider the converse to Theorem 1.1, and show it does not hold. In Theorem 6.3, we show that there exists a sequence of closed hyperbolic manifolds with bounded treewidth and volume approaching infinity. Thus, while volume gives an upper bound on treewidth, it does not give a lower bound.

On the other hand, recent work of Huszár, Spreer and Wagner [19] implies that there is a sequence of 3–manifolds whose treewidth approaches infinity. A corollary of our result is that any such examples that are hyperbolic have volume also approaching infinity. In fact, one family of examples is the family of small manifolds with large genus developed by Agol [1]. For the nth manifold in this family, combining work of [1] with [19], the treewidth is at least $\frac{1}{2}n$ but the volume is of order $O(n^2)$. It would be interesting to find a family for which volume and treewidth grow proportionally, to determine whether there is hope of improving Theorem 1.1.

1.1 Crushing, carving-width and treewidth

The proofs of Theorems 1.1 and 6.3 modify triangulations using the crushing procedure developed by Jaco and Rubinstein [20] and simplified by Burton [9]. In order to prove the theorems, we show in Corollary 3.4 that crushing does not affect a different measure of the sparsity of gluing relations of the triangulation, namely the *carving-width*; see Definition 2.2. For a 3–manifold triangulation, it is known that the carving-width is at least $\frac{2}{3}$ the treewidth and at most four times the treewidth; see Theorem 2.8. Thus crushing any finite number of times affects treewidth by at most a multiplicative constant.

Note that a usual pipeline for practical computation in 3–manifold topology consists of first simplifying a triangulation using efficient implementations of the crushing procedure and then running computations. Corollaries 3.4 and 3.5 guarantee that this approach does not affect the computational complexity of parametrised algorithms using the carving-width or treewidth as a parameter, such as [10; 11; 12]. Thus these results on crushing and computational complexity are important, and likely of independent interest.

Outline

In Section 2 we review results on treewidth and carving-width. Then, in Section 3, we prove that the Jaco–Rubinstein crushing procedure does not increase the carving-width of a triangulation. To prove Theorem 1.1, we use the fact that there is a universal Margulis constant μ such that any hyperbolic manifold M can be obtained from its μ–thick part $M^{\geq \mu}$ by hyperbolic Dehn filling; see Section 4.1. The proof begins by taking a geodesic triangulation of $M^{\geq \mu}$ with $O(\mathrm{vol}(M))$ *fat* tetrahedra, ie tetrahedra with volume bounded from below (Section 4). Next, we describe how to proceed to the Dehn filling without increasing the treewidth of the whole triangulation (Section 5). We consequently obtain a triangulation with treewidth $O(\mathrm{vol}(M))$, and describe an explicit algorithm to construct it. The number of tetrahedra of the triangulation depends solely on the hyperbolic volume $\mathrm{vol}(M)$ and the slopes of the hyperbolic Dehn surgeries.

Finally, in Section 6, we prove there exists a family of closed hyperbolic 3–manifolds with unbounded volume that admits a triangulation with constant treewidth.

2   Triangulations, carving-width and treewidth

In this section, we define several necessary terms and fix notation.

Let M be a closed 3–manifold. A *cell decomposition* of M is a pairwise-disjoint collection of n oriented, compact, convex linear 3–cells $\Delta_1, \ldots, \Delta_n$ equipped with affine maps that identify (or "glue together") their faces in pairs so that the underlying topological space is homeomorphic to M. The *dual graph* of a cell decomposition is the graph, with multiple arcs and loops, having a node for every 3–cell Δ_i and an arc (Δ_i, Δ_j) for every face gluing between 3–cells Δ_i and Δ_j.

A *generalised triangulation* of M is a cell decomposition where all 3–cells are abstract tetrahedra. Its dual graph is naturally a 4–valent graph, corresponding to gluings of

triangular faces. Generalised triangulations are widely used across major 3–manifold software packages, and they allow the representation of a rich variety of 3–manifolds using very few tetrahedra.

We also encounter *ideal triangulations* in this work, which are triangulations from which vertices have been removed. A removed vertex is called an *ideal vertex*.

Remark 2.1 We will be discussing both graphs and triangulations. We will refer to *nodes* and *arcs* of graphs, to clearly distinguish these from the *vertices* and *edges* of triangulations.

The *carving-width*, also known as *congestion*, is a graph parameter introduced by Seymour and Thomas [34].

Definition 2.2 Let $\mathcal{G}$ be a graph, possibly with loops and multiple arcs between nodes, defined on n nodes, and let T be an unrooted binary tree, with all internal nodes of degree 3 and with n leaves. An *embedding* π of $\mathcal{G}$ into T is an injective mapping from the nodes of $\mathcal{G}$ to the leaves of T. To every pair of endpoints (u, v) of an arc in $\mathcal{G}$ there corresponds a unique path $p(\pi(u), \pi(v))$ in T, connecting leaves $\pi(u)$ and $\pi(v)$. Define the *congestion* of an embedding π to be

$$\mathrm{cng}(\pi) = \max_{a \text{ arc of } T} |\{(u, v) \text{ in } \mathcal{G} : p(\pi(u), \pi(v)) \text{ contains } a\}|,$$

where note we count a multiple arc only once in the formula. Here $|\cdot|$ denotes the number of elements in a set.

The *carving-width* $\mathrm{cng}(\mathcal{G})$ of a graph $\mathcal{G}$ is the minimal congestion over all its embeddings into binary trees. The *carving-width of a cell decomposition* $\mathrm{cng}(\mathfrak{T})$ is the carving-width of its dual graph. Finally, we define the *carving-width of a 3–manifold M*, denoted by $\mathrm{cng}(M)$, to be the minimal carving-width over all its generalised triangulations.

Our definition of congestion differs from the literature by counting a multiple arc only once in the tree embedding. However, for a graph dual to a triangulation of a 3–manifold, this only affects the carving-width by a constant multiple, since all dual graphs of triangulations have constant maximal degree four. Also, note that a loop arc (u, u) leads to paths $p(\pi(u), \pi(u))$ of length 0 in a tree embedding, and can be disregarded when computing carving-width.

We give an example of a tree embedding in Figure 2. The following additional example will be important to our applications:
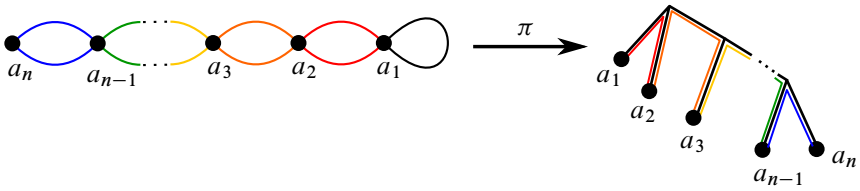
Figure 1: On the left is the daisy chain graph. Shown is an embedding into a tree, on the right, indicating that the carving-width is at most two.

Example 2.3 Let M be a solid torus. We describe a well-known triangulation of M, discussed in detail by Jaco and Rubinstein [21], called a *layered* triangulation or *layered solid torus*. At the core is a triangle with two sides identified to form a Möbius band. The first tetrahedron is glued so that two of its faces glue to the core triangle, one on either side of the triangle. By gluing correctly, the result is homeomorphic to a solid torus and has boundary consisting of exactly two triangles; see [21] for details. Additional tetrahedra may now be added inductively. At each step, a single tetrahedron is attached to the existing triangulation so that two of its faces are glued to the two boundary faces, covering a boundary edge (which then becomes an interior edge). The result is a solid torus with two triangular boundary faces. For any fixed slope on the torus, there exists a layered solid torus for which that slope is the meridian, ie bounds a disc; see for example [22, Theorem 4.1].

Consider now the gluing graph of a layered solid torus. The tetrahedron at the core with its two faces identified gives a loop at the corresponding node. The other two faces are identified to a single tetrahedron, giving two arcs to the next node. Any additional nodes are connected by two nodes to the previous node and two nodes to the next node. This forms a simple daisy chain. See Figure 1, left.

Note if there are only two nodes in the daisy chain graph, the carving-width is 1. In the more general case we obtain the following:

Lemma 2.4 *The daisy chain graph with $n \geq 3$ nodes, arising as the dual of a layered solid torus, has carving-width two.*

Proof The carving-width is at least two, because the carving-width of a graph is at least the maximal degree of a node after identifying multiple arcs, which is two for the daisy chain.

We now show that the carving width is at most two. Number the nodes of the daisy chain linearly, as in Figure 1, by $a_1, a_2, \ldots, a_{n-1}, a_n$, where the node a_1 has a loop

and two arcs running to a_2, the node a_2 has an additional two arcs running to a_3, etc, and a_n is 2–valent (two of its faces lie on the boundary of the layered solid torus).

Form an unrooted binary tree with n nodes as follows. Start with the unrooted binary tree T_3 with three leaves and a single 3–valent node. Label nodes b_1, b_2, b_3 in a cyclic manner. Note the two paths from b_1 to b_2 and from b_2 to b_3 run over the arc connecting the leaf b_2 exactly twice and other arcs exactly once. Now inductively increase the size of the tree until it has n leaves. Given an unrooted tree T_k with k leaves labelled $b_1, \ldots, b_k$ in a cyclic manner, form T_{k+1} by attaching two new leaves to the kth leaf, making it a 3–valent node, and label all leaves as in T_k except the two new leaves, labelled b_k and b_{k+1}, so that the labelling on T_{k+1} is still cyclic. See Figure 1, right.

Note that paths from b_i to b_{i+1} for $i < k - 1$ are identical in T_k and T_{k+1}. For $i = k - 1$ in T_{k+1}, the path from b_{k-1} to b_k runs over the arc connecting the leaf b_{k-1}, along the arc that connects the new 3–valent node in T_{k+1}, and then along the arc connecting the leaf b_k. The path from b_k to b_{k+1} runs over the two new arcs. Thus, by induction, all paths from b_i to b_{i+1} for $1 \leq i \leq k$ run over arcs connecting leaves $b_2, \ldots, b_k$ exactly twice and all other arcs exactly once. Continue until $k + 1 = n$, and map each a_i to b_i in T_n. Thus the carving-width is at most two. $\square$

A fundamental property of carving-width is that it decreases when taking *immersions*.

Definition 2.5 Let $\mathcal{G}$ be a graph, with adjacent arcs (u, v) and (v, w). A *lifting* of uvw consists of removing all arcs (u, v) and (v, w) from $\mathcal{G}$ and adding arc (u, w). An *immersion* of $\mathcal{G}$ is a graph $\mathcal{H}$ that can be obtained from $\mathcal{G}$ by a sequence of liftings and arc and node removals.

Equivalently, $\mathcal{H}$ is an immersion of $\mathcal{G}$ if there exists a mapping of the nodes of $\mathcal{H}$ to the nodes of $\mathcal{G}$ where every arc (u, v) is sent to a path from $\pi(u)$ to $\pi(v)$ in $\mathcal{G}$ such that distinct arcs in $\mathcal{H}$ lead to arc-disjoint paths in $\mathcal{G}$.

The following is standard and follows from the definitions:

Lemma 2.6 *If $\mathcal{H}$ is an immersion of $\mathcal{G}$, then*

$$\text{cng}(\mathcal{G}) \geq \text{cng}(\mathcal{H}).$$

Proof The nodes of $\mathcal{H}$ are a subset of the nodes of $\mathcal{G}$, so any embedding of $\mathcal{G}$ into a tree T restricts to an embedding of $\mathcal{H}$ into the same tree. Form $T_{\mathcal{H}}$ from T by

removing leaves of T that are not the image of nodes of $\mathcal{H}$, and viewing arcs adjacent to the remaining 2–valent nodes as a single arc. Then paths in $T_{\mathcal{H}}$ between nodes coming from $\mathcal{H}$ are obtained by taking paths in T between nodes of $\mathcal{H}$ and removing leaves and 2–valent nodes.

If $\mathcal{H}$ differs from $\mathcal{G}$ by a lifting of uvw, then $\mathcal{H}$ contains an arc (u, w) and no arcs (u, v) and (v, w), while $\mathcal{G}$ contains arcs (u, v) and (v, w). Note that the unique path in T from $\pi(u)$ to $\pi(w)$ can be obtained by taking the union of paths from $\pi(u)$ to $\pi(v)$ and from $\pi(v)$ to $\pi(w)$ and removing all arcs traversed twice. Thus the arcs in $(\pi(u), \pi(w))$ in T form a subset of those in the two paths $(\pi(u), \pi(v))$ and $(\pi(v), \pi(w))$. It follows that the congestion of $\mathcal{H}$ is at most that of $\mathcal{G}$.

Finally, for any node and arc removal to convert $\mathcal{G}$ to $\mathcal{H}$, the corresponding paths will be removed from T, so the number of paths $p(\pi(u), \pi(v))$ running over a fixed arc a in T, and hence $T_{\mathcal{H}}$, will also decrease. □

The carving-width of a graph is closely related to *treewidth*, which plays a major role in combinatorial algorithms. The *treewidth* of a graph was introduced by Robertson and Seymour [31], and is defined as follows:

Definition 2.7 Let $\mathcal{G}$ be a graph with loops and multiple arcs. A *tree decomposition* $(X, \{B_\tau\})$ of $\mathcal{G}$ consists of a tree X and *bags* B_τ of nodes of $\mathcal{G}$ for each node τ of X, for which

(1) each node u in $\mathcal{G}$ belongs to some bag B_τ;

(2) for every arc (u, v) in $\mathcal{G}$, there exists a bag B_τ containing both u and v;

(3) for every node u in $\mathcal{G}$, the bags containing u form a connected subtree of X.

The *width* of this tree decomposition is defined as $\max_{\tau \in X} |B_\tau| - 1$. The *treewidth of* $\mathcal{G}$, denoted by $\mathrm{tw}(\mathcal{G})$, is the smallest width of any tree decomposition of $\mathcal{G}$.

Similarly, the treewidth of a cell decomposition is the treewidth of its dual graph, and the treewidth of a 3–manifold is the minimal treewidth over all of its generalised triangulations.

Figure 2 shows the dual graph of a 9–tetrahedra triangulation of a 3–manifold, along with a possible tree decomposition. The largest bags have size three, and so the width of this tree decomposition is $3 - 1 = 2$.

Finally, treewidth and carving-width are closely related, and enjoy similar properties. First, they only differ by a constant multiplicative factor:
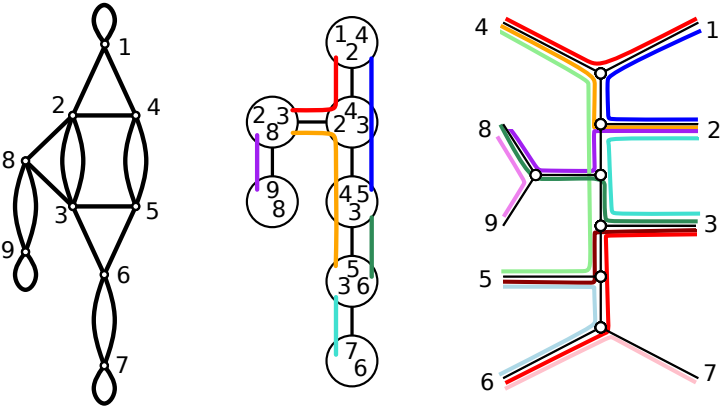
Figure 2: The dual graph of a 3–manifold triangulation (left), a tree decomposition of width 2 (centre) and a tree embedding of width 4 (right).

Theorem 2.8 [5, Theorem 1] *Let $\mathcal{G}$ be a graph of maximal degree d. Then*

$$\tfrac{2}{3}(\operatorname{tw}(\mathcal{G}) + 1) \leq \operatorname{cng}(\mathcal{G}) \leq d(\operatorname{tw}(\mathcal{G}) + 1).$$

Note that for dual graphs of generalised triangulations, the degree of every node is at most four, and treewidth and carving-width consequently differ by a small multiplicative constant.

The decision problem associated to computing the treewidth or carving-width of a graph is NP-complete [2; 34]. However, both treewidth and carving-width, together with an optimal tree decomposition or embedding into a tree, can be computed in time $O(f(k) \cdot n)$ on graphs with n nodes and treewidth/carving-width at most k [6; 35].

In the following, we use carving-width because of its favourable properties, and we connect it to the more widely used treewidth.

3 Crushing triangulations does not increase carving-width

This section focuses on compact manifolds with or without boundary, which are the main object of study of this article. However, all results cited and introduced extend naturally to ideal triangulations.

Crushing of triangulations is a fundamental technique introduced by Jaco and Rubinstein [20] to simplify 3–manifold triangulations. Let $\mathfrak{T}$ be a generalised triangulation of a 3–manifold M. A normal surface S in $\mathfrak{T}$ is a properly embedded surface in $\mathfrak{T}$
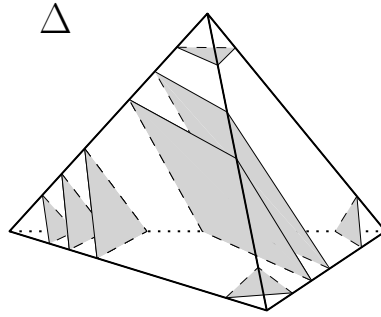
Figure 3: Tetrahedron cut by a normal surface. The intersection is a collection
of disjoint normal disks (quads and triangles).

that meets each tetrahedron in a (possibly empty) collection of curvilinear triangles
and quads, as illustrated in Figure 3. A *trivial surface* is a normal surface made only of
triangles; it always triangulates the link of a vertex. Finally, a 0–*efficient triangulation*
is a triangulation $\mathfrak{T}$ that either

- contains no nontrivial normal sphere if $\mathfrak{T}$ is closed or ideal, or

- contains no nontrivial normal disk if $\mathfrak{T}$ is bounded.

Crushing was introduced in [20] (see also [9]) as a means to construct 0–efficient
triangulations.

Definition 3.1 Let S be a normal surface in a triangulation $\mathfrak{T}$. *Crushing* the triangu-
lation along S consists of the following three steps:

(1) *Cut* $\mathfrak{T}$ open along S, leading to a cell decomposition with various cell types,
    presented in Figure 4.

(2) *Collapse* each copy of S to a point, using the quotient topology. This gives four
    types of cells: tetrahedra, 3–*sided footballs*, 4–*sided footballs* and *triangular
    purses* (all illustrated in Figure 4).

(3) *Flatten* all nontetrahedra cells to obtain a triangulation, ie flatten footballs into
    edges and triangular purses into triangles, as in Figure 4.

Conclude by separating tetrahedra joined by pinched vertices and edges.

Following [9, Lemma 3], the flattening step can be performed iteratively, one non-
tetrahedron cell at a time. In particular, flattening a football or a triangular purse induces
the flattening of bigonal faces in the adjacent cells, hence creating temporary cells of
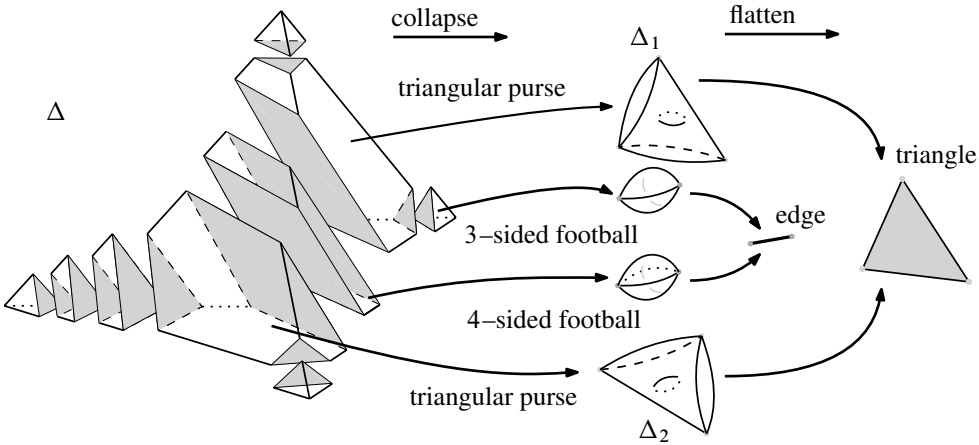
Figure 4: Cut-out tetrahedron containing quads, collapsing of the cells and flattening. The collapsing produces two triangular purses and a collection of 3– and 4–sided footballs. Footballs are flattened into edges and triangular purses into triangles.

new types: triangular purses with one or two flattened bigons (also known as bigonal pyramid and triangular pillows, respectively) and 2–sided footballs (also known as bigonal pillows).

In particular, we use the following property:

Theorem 3.2 (Jaco and Rubinstein [20]; see also Burton [9]) *Let $\mathfrak{T}$ be a generalised triangulation of a closed or bounded 3–manifold M. There is an algorithm to construct a finite family of triangulations $\mathfrak{T}_1, \ldots, \mathfrak{T}_n$ triangulating manifolds $M_1, \ldots, M_n$ such that $M = M_1 \# \cdots \# M_n$ and each $\mathfrak{T}_i$ is either 0–efficient or can be shown to be a triangulation of S^3, $S^2 \times S^1$, $\mathbb{R}P^3$ or $L(3,1)$. The algorithm consists of finding normal spheres and disks in the original triangulation, and crushing them.*

Jaco and Rubinstein prove that 0–efficient triangulations of a closed irreducible manifold have one vertex, and 0–efficient triangulations of a bounded irreducible ∂–irreducible manifold (without 2–sphere boundary components[1]) have all vertices in the boundary, with exactly one vertex per boundary component.

We now introduce the main combinatorial result of this article, namely that crushing does not increase the carving-width. We use this result repeatedly as a tool to manipulate hyperbolic manifolds in the latter sections.

---

[1]This is a technicality that only rules out the simple case where the irreducible manifold is a 3–cell.

Theorem 3.3 Let $\mathfrak{T}$ be a generalised (or ideal, or bounded) triangulation of a 3–manifold M, and S a normal surface in $\mathfrak{T}$. Let $\mathfrak{T}^*$ be the triangulation obtained after crushing S. Then the dual graph of $\mathfrak{T}^*$ is an immersion of the dual graph of $\mathfrak{T}$.

Proof We track the evolution of the cell decomposition of the triangulation under the three steps (cut, collapse, flatten) of crushing S in $\mathfrak{T}$, in order to describe the change to its dual graph. Let $\mathcal{G}$ be the dual graph of $\mathfrak{T}$.

Cut $\mathfrak{T}$ along S. The result is a collection of cells; each tetrahedron that meets S in $\mathfrak{T}$ is split into cells across normal discs of S as in Figures 3 and 4.

Now collapse each normal disc of S to a point, using the quotient topology, to obtain the cell complex $\mathfrak{T}'$. This operation splits every tetrahedron Δ of $\mathfrak{T}$ into a collection $C_\Delta = \{\Delta_0, \ldots, \Delta_n\}$ of cells, where n is the number of normal disks in $\Delta \cap S$. The cells are of four types: tetrahedra, 3–sided footballs, 4–sided footballs and triangular purses; see Figure 4.

Note that if $\Delta \cap S$ contains no quad, then C_Δ is made of exactly one (central) tetrahedron and a possibly empty collection of 3–sided footballs. If $\Delta \cap S$ contains quads, then C_Δ is made of two triangular purses and a possibly empty collection of 3–sided and 4–sided footballs.

Now flatten. We obtain a generalised triangulation $\mathfrak{T}^*$. The 3– and 4–sided footballs become edges, and thus have no dual nodes or arcs in $\mathfrak{T}^*$. A tetrahedron is not flattened, thus the dual graph of $\mathfrak{T}^*$ has one node corresponding to $\Delta \cap S$ containing no quads. Two triangular purses flatten to triangles, thus removing the node corresponding to Δ.

Note if we perform this process one tetrahedron at a time, adjusting the dual graph one node at a time, then each node corresponding to Δ that does not meet a quad will be replaced by a node corresponding to a tetrahedron of the crushing; the arcs from this node will run to the same nodes as before the replacement, since faces of the new tetrahedron are still glued to faces of adjacent tetrahedra. For each node corresponding to Δ that meets a quad, the node is removed, and two liftings are performed: when each triangular purse is flattened, it identifies two triangular faces together and removes a node. Thus faces of adjacent tetrahedra become glued through this triangle. The result is a lifting. See Figure 5.

Perform this process for each tetrahedron. Then separate tetrahedra joined by pinched vertices and edges, which does not affect the dual graph. We see that the final result is an immersion. $\qquad\square$
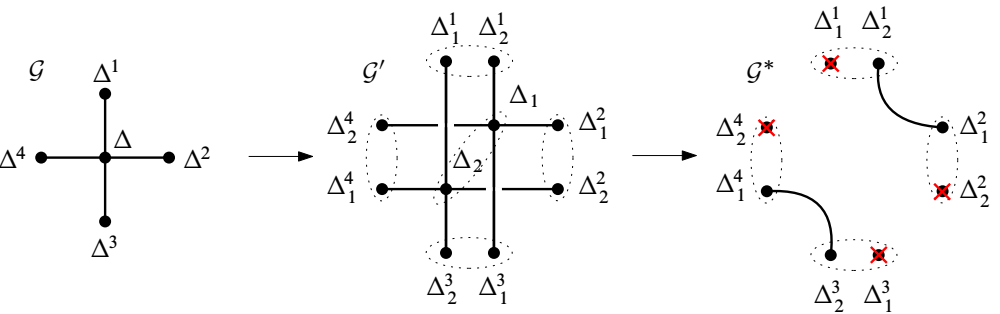
Figure 5: Local transformation of the dual graph at tetrahedron Δ from Figure 4 when crushing iteratively at Δ. $\mathcal{G}$ is the dual graph before crushing, $\mathcal{G}'$ is the dual graph after cutting, collapsing and then flattening only the 3– and 4–sided footballs from Δ (Δ_1 and Δ_2 stand for the two triangular purses), and $\mathcal{G}^*$ is the dual graph after flattening the triangular purses. Note that some of the nodes Δ_i^j in $\mathcal{G}'$ may have already been removed when flattening adjacent footballs, and some of the bigon faces of triangular purses may already be collapsed. This does not change the analysis as it only removes nodes and arcs from the dual graph. $\mathcal{G}^*$ is obtained from $\mathcal{G}$ by lifting $\Delta^1 \Delta \Delta^2$ and $\Delta^3 \Delta \Delta^4$, then removing the node Δ. Because their corresponding cells have bigonal faces, and hence cannot be tetrahedra, the crossed-out nodes on $\mathcal{G}^*$ will be removed from the graph when flattening adjacent cells. The immersion of $\mathcal{G}^*$ into $\mathcal{G}$ is obtained by mapping the (nonremoved) nodes Δ_i^j in $\mathcal{G}^*$ to Δ^j in $\mathcal{G}$.

Corollary 3.4 *Crushing does not increase carving-width.*

Proof Immersion does not increase carving-width, by Lemma 2.6. □

Corollary 3.5 *Crushing an arbitrary finite number of normal surfaces in a triangulation increases the treewidth by at most a multiplicative factor of six.*

Proof This follows from Corollary 3.4 and Theorem 2.8, using the fact that a graph coming from the dual of a 3–manifold triangulation has all nodes of degree at most four. □

Remark 3.6 In general the two liftings pictured in Figure 5 may increase the treewidth of a graph. For example, consider the graphs in Figure 6, where the graph of the pentagonal prism (on the right) is obtained from liftings 2–0–10 and 1–0–9 in the graph on the left. The latter has treewidth 3, as it admits K_4 as a minor (tw ≥ 3), and a path decomposition $(B_i)_{i=1,\dots,8}$, $B_i = \{0, i, i+1, i+2\}$, of width 3. The former is a well-known obstruction to treewidth 3, and has treewidth 4 [3; 33].
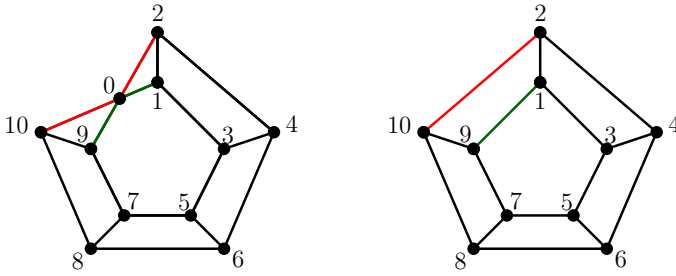
Figure 6: Liftings of 2–0–10 and 1–0–9 increase the treewidth of the graph.

In their seminal work on 0–efficient triangulations, Jaco and Rubinstein proved that a minimal triangulation (ie with a minimal number of tetrahedra) of a manifold is 0–efficient. In the same spirit, we deduce the following for triangulation width:

Corollary 3.7 *Any closed, orientable, irreducible 3–manifold M, not S^3, $S^2 \times S^1$, $\mathbb{R}P^3$ or $L(3,1)$, admits a 0–efficient triangulation of optimal carving-width $\mathrm{cng}(M)$.*

Any compact, orientable, irreducible, ∂–irreducible 3–manifold M, not the 3–cell, admits a 0–efficient triangulation of optimal carving-width $\mathrm{cng}(M)$.

Proof Let $\mathfrak{T}$ be a triangulation of M of carving-width $\mathrm{cng}(M)$. By Theorem 3.2, one can crush normal spheres and disks in $\mathfrak{T}$ to get a 0–efficient triangulation $\mathfrak{T}^*$ of carving-width at most $\mathrm{cng}(M)$ by Corollary 3.4.   □

To conclude this section, we prove the following simple property of carving-width:

Lemma 3.8 *Let $\mathcal{G}$ and $\mathcal{G}'$ be two graphs, and let $\mathcal{G} \# \mathcal{G}'$ be obtained by adding $m \geq 1$ arcs between nodes of $\mathcal{G}$ and nodes of $\mathcal{G}'$, not counting multiplicities. Then*

$$\mathrm{cng}(\mathcal{G} \# \mathcal{G}') \leq \max\{\mathrm{cng}(\mathcal{G}) + m - 1, \, \mathrm{cng}(\mathcal{G}') + m - 1, \, \max \text{ degree in } \mathcal{G} \# \mathcal{G}'\}.$$

If the m arcs are incident to a single node u in $\mathcal{G}$, then

$$\mathrm{tw}(\mathcal{G} \# \mathcal{G}') \leq \max\{\mathrm{tw}(\mathcal{G}), \mathrm{tw}(\mathcal{G}') + 1\}.$$

Proof Pick two optimal tree embeddings $\pi \colon \mathcal{G} \to T$ and $\pi' \colon \mathcal{G}' \to T'$, with arcs $a \in T$ and $a' \in T'$ realising the congestion $\mathrm{cng}(\pi)$ and $\mathrm{cng}(\pi')$, respectively. Without loss of generality, suppose $\mathrm{cng}(\pi) \geq \mathrm{cng}(\pi')$; hence, at least as many paths run through a as a'. Let u in $\mathcal{G}$ and v in $\mathcal{G}'$ be two nodes that are adjacent in $\mathcal{G} \# \mathcal{G}'$. Subdivide the only arcs incident to the leaves of $\pi(u)$ and $\pi'(v)$ in the tree embeddings, and connect the two new nodes by an arc. This leads to a tree embedding $\Pi(\mathcal{G} \# \mathcal{G}')$ of $\mathcal{G} \# \mathcal{G}'$.

Note that m paths run through the arc connecting the trees T and T'. The congestion $\mathrm{cng}(\Pi)$ will be largest if as many of those paths as possible also run through a. If a in T does not connect the leaf $\pi(u)$, then at most $m - 1$ new paths run through a, because the path from $\Pi(u)$ to $\Pi(v)$ only runs through new arcs.

If a in T is the arc connecting the leaf $\pi(u)$, then $\mathrm{cng}(\mathcal{G})$ paths run to u, hence u is $\mathrm{cng}(\mathcal{G})$–valent. The arc a is subdivided to form the new tree, and $\mathrm{cng}(\mathcal{G})$ paths from the tree T will continue to run over the two new arcs obtained by subdividing a. If one of the m new arcs between $\mathcal{G}$ and $\mathcal{G}'$ does not have an endpoint on u, then the corresponding path will run over the subarc of a that does not meet the leaf of $\Pi(u)$, whereas the arc from $\Pi(u)$ to $\Pi(v)$ will not meet this arc, and thus the congestion is at most $\mathrm{cng}(\mathcal{G}) + m - 1$. However, if all the m new arcs between $\mathcal{G}$ and $\mathcal{G}'$ run from u to nodes of $\mathcal{G}'$, then all m new paths must also run over the arc connecting the leaf of $\Pi(u)$. Thus the congestion will be at most $\mathrm{cng}(\mathcal{G}) + m$, which is the degree of u in $\mathcal{G} \# \mathcal{G}'$.

For treewidth, pick two optimal tree decompositions T and T' for $\mathcal{G}$ and $\mathcal{G}'$, respectively, and let u in $\mathcal{G}$ be the node to which all new arcs are incident. Let (u, v), with v in $\mathcal{G}'$, be a new arc. Let B_u be a bag of T containing u and B_v be a bag of T' containing v. Connecting B_u and B_v with an arc, and adding node u to all bags in T', leads to a tree decomposition of $\mathcal{G} \# \mathcal{G}'$, and the result follows. $\qquad\square$

4 Triangulating thick hyperbolic manifolds

In this section, we consider a finite-volume, compact, hyperbolic 3–manifold M with boundary and bounded injectivity radius. We show such a manifold admits a triangulation with a bounded number of tetrahedra, where the bound is linear in volume. This result is not new; its proof is outlined in Thurston's notes [36], and proved carefully elsewhere, for example by Kobayashi and Rieck [25]. We step through highlights of the proof here for completeness, and also to discuss the algorithmic nature of the argument, in order to actually compute a triangulation. The algorithm will be summarised in Section 5.1.

4.1 Hyperbolic manifolds

Here we review definitions and results in hyperbolic geometry that are most important to our results. For further information on hyperbolic 3–manifolds, see for example [4].

Definition 4.1 Let $\mu > 0$ and let M be a hyperbolic 3–manifold. The μ–*thick part* *of* M, denoted by $M^{\geq \mu}$, consists of all points $x \in M$ such that any geodesic based at x has length at least μ. Equivalently, $M^{\geq \mu}$ is the set of points in M with injectivity radius at least $\frac{1}{2}\mu$.

The complement of the μ–thick part is the μ–thin part.

Recall that, by the Margulis lemma, there exists a universal constant ϵ_3 such that for any finite-volume, hyperbolic 3–manifold M and any $\mu \leq \epsilon_3$, the μ–thin part of M consists only of tubes about geodesics and cusps [23]. In the discussion below, we will always assume that $0 < \mu \leq \epsilon_3$. Such a μ is said to be a 3–dimensional *Margulis constant*.

Let $B_D(x, r)$ denote the open ball of centre x and radius $r > 0$ in D. Recall that the volume of a hyperbolic 3–ball of radius $r > 0$ is given by

$$\text{vol}(B_{\mathbb{H}^3}(r)) = \pi(\sinh 2r - 2r);$$

see for example [16].

4.2 Triangulating thick parts

In this section, for a fixed 3–dimensional Margulis constant μ, we recall the argument of [25] to show that a small neighbourhood of the μ–thick part of a hyperbolic 3–manifold M can be triangulated with $O(\text{vol}(M))$ tetrahedra.

We start by setting notation. For $\mu > 0$ a Margulis constant and any $d > 0$, denote the metric d–neighbourhood of $M^{\geq \mu}$ by $X := N_d(M^{\geq \mu})$. In [25, Proposition 1.2], it is shown that there exists $R := R(\mu, d)$ such that for any complete finite-volume, hyperbolic 3–manifold M and any $x \in X$, the injectivity radius of x is at least R, and X is obtained from M by drilling out short geodesics and truncating cusps. Let $D = \min\{R, d\}$.

Definition 4.2 Let (X, dist_X) be a metric space. For $\varepsilon > 0$, a set of points $P \subset X$ is ε–*dense* in X if, for any $x \in X$, there is a point $p \in P$ such that $\text{dist}_X(x, p) < \varepsilon$. For $1 \geq \delta > 0$, the set P is $\delta \varepsilon$–*separated* if any two distinct points $p, q \in P$ satisfy $\text{dist}_X(p, q) \geq \delta \varepsilon$. We call P a (δ, ε)–*net* if it is ε–dense and $\delta \varepsilon$–separated. Note that any (δ, ε)–net is also a (δ', ε)–net for any $\delta' \leq \delta$.

Lemma 4.3 *Let* $\mu > 0$ *be a 3–dimensional Margulis constant and* $d > 0$. *Let* M *be a hyperbolic manifold of finite volume* $\text{vol}(M)$, *with* d–*neighbourhood of the thick*

part $M^{\geq \mu}$ denoted by X. Then X admits a (δ, ε)–net of size

$$n \leq \frac{\mathrm{vol}(M)}{\pi (\sinh \varepsilon - \varepsilon)} \leq \frac{6}{\pi} \frac{\mathrm{vol}(M)}{\varepsilon^3}$$

for any $\varepsilon \leq \mu$ and any $\delta \leq 1$.

Proof This is the standard iterative construction of nets. Fix an arbitrary $\varepsilon \leq \mu$. Set P to be the empty set. While there exists a point x in the set

$$X - \bigcup_{p \in P} B_X(p, \varepsilon),$$

set P to be $P \cup \{x\}$. At any time of the procedure, the union of balls of radius $\frac{1}{2}\varepsilon$ centred on the points of P are disjoint and embedded in M. Consequently,

$$|P| \times \mathrm{vol}\big(B_{\mathbb{H}^3}\big(\tfrac{1}{2}\varepsilon\big)\big) \leq \mathrm{vol}(M).$$

Because M has finite volume, the procedure terminates, and P is a $(1, \varepsilon)$–net for X by construction. $\qquad \square$

Recall that $D = \min\{R, d\}$. Kobayashi and Rieck take a maximal D–separated set for X, but it suffices for their argument to let $\{x_1, \ldots, x_N\}$ be a $(1, D)$–net for X. Now let $\{V_1, \ldots, V_N\}$ be the *Voronoi cells* in M corresponding to $\{x_1, \ldots, x_N\}$, namely the sets

$$V_i = \{p \in M \mid \mathrm{dist}(p, x_i) \leq \mathrm{dist}(p, x_j) \text{ for } j = 1, \ldots, N\}.$$

Kobayashi and Rieck show that the components of $V_i \cap X$ consist of handlebodies with universally bounded genus, with boundaries consisting of geodesic faces meeting in geodesic edges and vertices, and that the number of such faces and edges (and vertices) is universally bounded independent of M.

After possibly perturbing the points $\{x_1, \ldots, x_N\}$ slightly, they give an algorithm that builds, for each component $V_{i,j}$ of $V_i \cap X$, a 2–complex $K_{i,j}$. The complex $K_{i,j}$ has totally geodesic faces, a universally bounded number of faces and edges, and it cuts $V_{i,j}$ into a single ball $B_{i,j}$.

By subdividing remaining faces into triangles and then coning to the centre of the ball $B_{i,j}$, we obtain a triangulation of $V_{i,j}$ such that, by construction, triangulations of distinct $V_{i,j}$ agree on their intersections.

The above gives the following, which is [25, Proposition 1.4]:

Proposition 4.4 Let μ be a 3–dimensional Margulis constant and fix $d > 0$. For any complete finite-volume, hyperbolic 3–manifold M, let X denote the metric d–neighbourhood of $M^{\geq \mu}$. Then there exists a constant $C = C(\mu, d)$ such that the following holds:

(1) M is decomposed into $N \leq C \operatorname{vol}(M)$ Voronoi cells $\{V_1, \dots, V_N\}$.

(2) $V_i \cap X$ is triangulated using at most C tetrahedra for all $i = 1, \dots, N$.

(3) For any $i, j \in \{1, \dots, N\}$, the triangulations in (2) coincide on the intersection $(V_i \cap X) \cap (V_j \cap X)$.

We now obtain the following consequence, which is [25, Theorem 1.1]:

Theorem 4.5 Let μ be a 3–dimensional Margulis constant and fix $d > 0$. Then there exists a constant $v(\mu, d) > 0$ such that for M any closed hyperbolic 3–manifold with volume $\operatorname{vol}(M)$, the metric d–neighbourhood of the μ–thick part $M^{\geq \mu}$ admits a triangulation $\mathfrak{T}_B$ with number of tetrahedra at most

$$\frac{\operatorname{vol}(M)}{v(\mu, d)} = O(\operatorname{vol}(M)).$$

Proof Apply Proposition 4.4 to M, decomposing X into at most C cells with at most C tetrahedra each. Because the triangulations match where the cells overlap, this gives a triangulation of $N_d(M^{\geq \mu})$ with at most $C^2 \operatorname{vol}(M)$ tetrahedra. Set $v(\mu, d) = 1/C^2$. □

Naturally, this triangulation has carving-width (and treewidth) at most $O(\operatorname{vol}(M))$.

5 Triangulations of closed manifolds

For μ a Margulis constant and fixed $d > 0$, let $M^{\geq \mu}$ denote the thick part of M and let $X := N_d(M^{\geq \mu})$ denote its d–neighbourhood. Then X is a compact manifold with (possible) torus boundary components; see [25, Proposition 2.1]. Let $\mathfrak{T}_B$ be the triangulation of X constructed in Theorem 4.5. This triangulation contains at most $\operatorname{vol}(M)/v(\mu, d)$ tetrahedra.

Lemma 5.1 There exists a triangulation $\mathfrak{T}_{JR}$ of X that has exactly one vertex in each boundary component of $M^{\geq \mu}$, no other vertices, and contains at most $\operatorname{vol}(M)/v(\mu, d)$ tetrahedra.

Proof For the proof, we wish to obtain a 0–efficient triangulation, as in work of Jaco and Rubinstein [20], repeated in Theorem 3.2. To obtain a 0–efficient triangulation

from $\mathfrak{T}_B$, apply the crushing procedure. Crushing does not increase the number of tetrahedra. However, it may affect the topology of the underlying manifold, but only in well-understood ways that are listed in [9, Theorem 2]: it may

- undo connect sums,
- cut open along properly embedded discs,
- fill a boundary sphere with a 2–ball, or
- delete a 3–ball, 3–sphere, $\mathbb{R}P^3$, $L(3, 1)$ or $S^2 \times S^1$ component.

Since we start with M hyperbolic, the interior of the manifold $M^{\geq \mu}$ is also hyperbolic. Hence there are no connect sums, no boundary spheres and no nonhyperbolic components. There are also no essential discs; thus if we cut along a properly embedded disc, the disc will cut off a ball, which we may ignore.

Thus repeatedly applying the crushing move gives a 0–efficient triangulation, which has the properties required by the lemma. □

Lemma 5.2 *If $\mathfrak{T}$ is a triangulation of a 3–manifold M with a torus boundary component S such that S inherits a one-vertex triangulation from $\mathfrak{T}$, then any Dehn filling of M along S can be given a triangulation with carving-width at most*

$$\max\{\mathrm{cng}(\mathfrak{T}) + 1, 4\}.$$

Proof First, note that the only one-vertex triangulation of the torus S has two triangles, so the boundary component of $\mathfrak{T}$ corresponding to S has two triangles, which are incident to at most two distinct tetrahedra.

The Dehn filling can be obtained by attaching a layered solid torus, as in Example 2.3, with gluing graph a simple daisy chain, with one loop arc with both endpoints on the node corresponding to the tetrahedron in the centre, and pairs of parallel arcs between nodes corresponding to the layers of tetrahedra. The layering is determined by the slope of the Dehn filling. After a finite number of steps, the meridian of the layered solid torus will correspond to the desired slope; see [21]. At this stage, the final pair of faces is attached to the tetrahedra in M that form the torus boundary.

Because the triangulation of M and the one of the layered solid torus are glued along two faces of at most two distinct tetrahedra, the carving-width after gluing is at most the maximum of the carving-width between the two triangulations plus 1, or the maximal degree in the dual graph after filling, which is 4, by Lemma 3.8. Because a daisy chain graph has carving-width at most two, by Lemma 2.4, the result follows. □

Theorem 5.3 Let μ be a 3–dimensional Margulis constant and fix $d > 0$. Let M be a closed hyperbolic 3–manifold with volume $\mathrm{vol}(M)$. Then the carving-width of M is bounded above by $6 \cdot \mathrm{vol}(M)/v(\mu, d) = O(\mathrm{vol}(M))$.

Proof Start with the triangulation $\mathfrak{T}_{\mathrm{JR}}$ of $X := N_d(M^{\geq \mu})$. The triangulation $\mathfrak{T}_{\mathrm{JR}}$ is obtained in Lemma 5.1 by crushing, and by Corollary 3.4 has carving-width at most the carving-width of triangulation $\mathfrak{T}_B$ in Theorem 4.5, which is naturally at most $4 \cdot \mathrm{vol}(M)/v(\mu, d) = O(\mathrm{vol}(M))$.

We obtain M from X by Dehn filling the torus boundary components. Each boundary component inherits a triangulation from $\mathfrak{T}_{\mathrm{JR}}$ with exactly one vertex and two triangles, and so there are at most $2 \cdot \mathrm{vol}(M)/v(\mu, d)$ boundary components. Consequently, performing the Dehn fillings increases the carving-width by at most $2 \cdot \mathrm{vol}(M)/v(\mu, d)$, by Lemma 5.2. □

By Theorem 2.8, this result also applies to treewidth.

5.1 Algorithm and computational complexity

Our approach to constructing a triangulation with carving-width $O(\mathrm{vol}(M))$ for a closed hyperbolic 3–manifold M is algorithmic. Given M presented by its thick part and hyperbolic Dehn fillings, and an oracle to access its geometry, one can compute the $(1, D)$–net of Lemma 4.3 with $O(\mathrm{poly}(\mathrm{vol}(M)))$ calls to the oracle, where poly is a polynomial function. The procedure to compute Voronoi cells is polynomial in the size of the net, which is $O(\mathrm{vol}(M))$. Meshing in polynomial time, while maintaining the Voronoi cells, can be done using an intrinsic version of Delaunay refinement [15]. Kobayashi and Rieck's algorithm to subdivide each cell $V_{i,j}$ into a ball by constructing a 2–complex $K_{i,j}$ requires first building a graph with desirable properties, in [25, Lemma 4.2], which is constructed by adding components of $V_{i,j} \cap \partial X$ one at a time and then adding and removing edges that lie in geodesic faces. The number of components of $V_{i,j} \cap \partial X$ is universally bounded, depending on the universal bound of the genus of $V_{i,j}$, and thus for each $V_{i,j}$ this algorithm runs in constant time. One can then triangulate the balls $V_{i,j} - K_{i,j}$ in constant time, using the fact that the number of faces, edges and vertices of each ball is universally bounded. This is done for each of the $O(\mathrm{vol}(M))$ cells.

The algorithm using the crushing procedure in the proof of Lemma 5.1 is exponential in $\mathrm{vol}(M)$, the number of tetrahedra. However, the complexity becomes polynomial [9] if, instead of 0–efficient triangulations, we only require the triangulations $\mathfrak{T}_i$ of the

decomposition described in Theorem 3.2 to have exactly one vertex per boundary component if bounded, or one vertex triangulation if closed. This is sufficient for the construction of Theorem 5.3.

The filling by Dehn surgery in the proof of Theorem 5.3 is linear time in the Dehn filling coefficients (and hence linear in the final number of tetrahedra for the output triangulation). Consequently, the procedure above constructs a triangulation with treewidth vol(M) for a closed hyperbolic 3–manifold M in time

$$O\big(\mathrm{poly}(\mathrm{vol}(M)) \cdot \mathrm{Or} + \mathrm{poly}(\mathrm{vol}(M)) + n\big)$$

and $O(n)$ memory, where poly denotes polynomial functions, Or is the time complexity for calling the geometry oracle and n is the number of tetrahedra of the output manifold. Note that, by construction, we also get a tree-decomposition of width $O(\mathrm{vol}(M))$.

6 Treewidth does not bound volume

In this section, we prove there exist families of manifolds with constant treewidth but unbounded volume. Our examples include both manifolds with boundary and closed manifolds.

The manifolds with boundary that we consider are the exteriors of 2–bridge knots. There are many ways to describe 2–bridge knots; see for example [7]. For the purpose of this paper, a 2–bridge knot $K[a_{n-1}, \ldots, a_1]$ is described by a finite collection of integers $a_1, \ldots, a_{n-1}$. The diagram of the knot $K[a_{n-1}, \ldots, a_1]$ consists of $n-1$ twist regions arranged linearly, with the ith region containing $|a_i|$ crossings, and the direction of the crossing determined by the sign of a_i, and twist regions connected as shown in Figure 7. In general, we may always assume that the a_i are either all positive or all negative, and $|a_1|$ and $|a_{n-1}|$ are at least 2.

Proposition 6.1 *The family of hyperbolic 2–bridge knots has unbounded volume but treewidth bounded by a constant. This is true whether we take treewidth corresponding to an ideal triangulation or corresponding to a finite triangulation of the knot exterior.*

Proof By Theorem B.3 of [18], the volume of a 2–bridge knot $K[a_{n-1}, \ldots, a_1]$ is bounded below by $2v_3 n$, where $v_3 = 1.0149\ldots$ is a constant. Thus letting n approach infinity gives a sequence of 2–bridge knots with volume approaching infinity.
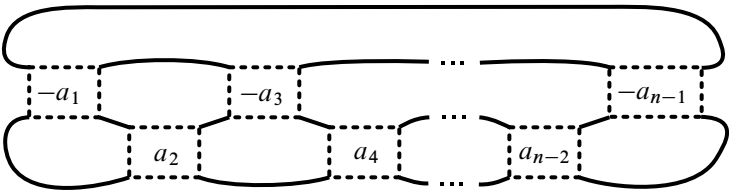
Figure 7: The diagram of $K[a_{n-1}, \ldots, a_1]$ for n even. The box labelled $\pm a_i$ denotes a horizontal twist region with $|a_i|$ crossings, with the sign of all crossings equal to the sign of $\pm a_i$. The crossing number is $C = |a_{n-1}| + \cdots + |a_1|$.

To show that these knots have bounded treewidth, we need to describe a triangulation of the knot complements. We use the well-known triangulation of 2–bridge knot complements due to Sakuma and Weeks [32]; see also Guéritaud and Futer [18]. We will review the description of ideal triangulation briefly here. More details can be found in the two previous references, or in [30].

For ease of exposition, we will only work with examples for which n is even, each a_i is positive and a_1 and a_{n-1} are both at least 2. While a similar argument works when n is odd or when the values a_i are all negative, we will not need it for our purposes here.

The easiest way to describe the triangulation is to start with the diagram of the knot $K[a_{n-1}, \ldots, a_1]$ as in Figure 7, and then isotope all the odd twist regions to be vertical, as in Figure 8.

With the diagram in the form of Figure 8, we may think of the crossings as nested, with the first crossing of the twist region corresponding to the first crossing of a_1 on the inside and the last crossing corresponding to the last crossing of a_{n-1} on the outside.
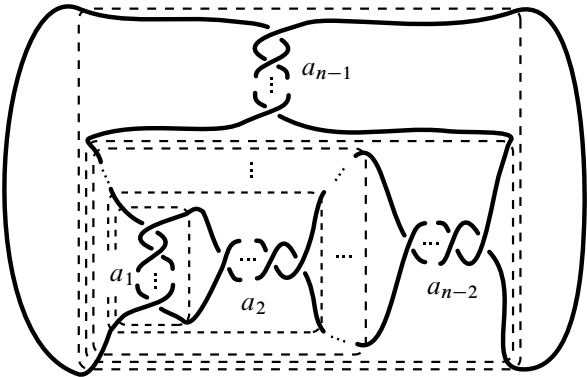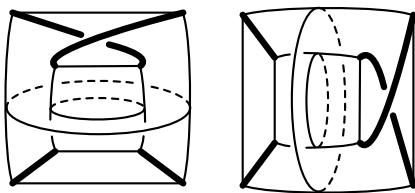

Figure 8: Another diagram of a 2–bridge knot.

Figure 9: Vertical (left) and horizontal (right) blocks of the form $S \times I$. The 4–punctured spheres on the outside and inside correspond to $S \times \{1\}$ and $S \times \{0\}$, respectively. Figure from [30].

Recall that there are $C = a_1 + \cdots + a_{n-1}$ crossings in total. The complement of the knot is built of the following:

(1) A tangle containing the very first crossing on the inside.

(2) For the ith crossing, for $i = 2, \ldots, C-1$, a block homeomorphic to $S \times I$, where S is a 4–punctured sphere. The block contains a single horizontal or vertical crossing, depending on whether the ith crossing is horizontal or vertical. See Figure 9. It is stacked onto the previous crossing.

(3) A tangle containing the very last crossing on the outside.

There are $2(C-2)$ tetrahedra in the decomposition, which we now describe. First, ignore the innermost and outermost tangles containing a single crossing. For the ith crossing, for $i = 2, \ldots, C-2$, there is a pair of tetrahedra lying between the ith and $(i+1)^{\text{st}}$ blocks $S \times I$. The ideal edges of these tetrahedra are edges that are either horizontal or vertical on one of the surfaces $S \times \{0\}$ and $S \times \{1\}$, for the ith or $(i+1)^{\text{st}}$ block. When we isotope all these edges to lie between the two blocks, we see the form of the two tetrahedra, as in Figure 10, left. One tetrahedron, denoted by T_i^1 lies in front of the ith block, and one, denoted by T_i^2, lies behind.

We need to determine how the faces of the tetrahedra glue. Note that the faces on the outside are isotopic through the outside block to triangles on the outside block. Consider the two outer faces of T_i^1 on $S \times \{0\}$ on the $(i+1)^{\text{st}}$ block. These two faces are shown in Figure 10 for the example that the crossing is vertical. When we isotope through the $(i+1)^{\text{st}}$ block, the triangles isotope to the triangles shown on the right of Figure 10. Note that one lies in front and one lies in the back. The one in front will be glued to an inside face of T_{i+1}^1, and the one in the back will be glued to an inside face of T_{i+1}^2. Similarly, if the crossing in the $(i+1)^{\text{st}}$ block is horizontal, triangle faces of T_i^1 on the outside isotope to two triangles, one on the front and one on the back of $S \times \{1\}$ in the $(i+1)^{\text{st}}$ block. Thus one will be glued to T_{i+1}^1 and the other to T_{i+1}^2.
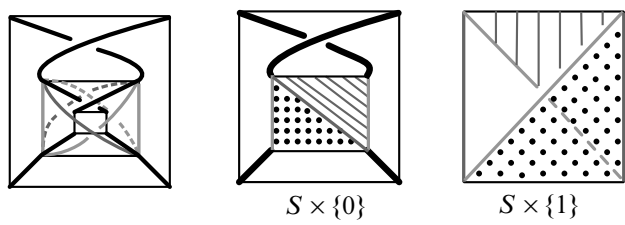
Figure 10: Left: the edges of the tetrahedron are shown. Middle: two faces of the tetrahedron T_i^1 lying on the surface $S \times \{0\} \subset S \times I$ for the $(i+1)^{\text{st}}$ block. Right: position of those two faces when isotoped to $S \times \{1\}$ on the $(i+1)^{\text{st}}$ block. Figure from [30].

An identical argument, isotoping in the other direction, then implies that one of the inside faces of T_i^1 is glued to a face of T_{i-1}^1 and the other inside face is glued to T_{i-1}^2.

To complete the dual graph of the triangulation, we must determine how inner faces of T_2^1 and T_2^2 glue when we insert the innermost tangle containing a single crossing. As described in [18], inserting this tangle glues the two faces on the block $S \times I$ containing the second crossing to each other. The two triangles in the front of $S \times \{0\}$ are glued, and the two triangles in the back of $S \times \{0\}$ are glued. However, recall that we view tetrahedra T_2^1 and T_2^2 as lying between the 2nd and 3rd blocks. As in Figure 10, right, isotoping the inner faces of T_2^1 and T_2^2 through the block $S \times I$ to $S \times \{0\}$ puts the two faces of T_2^1 on opposite sides of $S \times \{0\}$. Thus when we attach the tangle containing the first crossing, we glue each inner face of T_2^1 to an inner face of T_2^2. A similar argument shows each outer face of T_{C-2}^1 is glued to an outer face of T_{C-2}^2.

Thus the graph dual to the triangulation of the 2–bridge knot complement has the form shown in Figure 11.

A tree decomposition of the graph of Figure 11 is shown in Figure 12. There are $C - 4$ bags in the tree decomposition. Each bag contains exactly four nodes, namely T_i^1, T_i^2, T_{i+1}^1 and T_{i+1}^2, for $i = 2, 3, \ldots, C-3$. Thus the treewidth of this tree decomposition is $4 - 1 = 3$. Note it is constant, independent of the values of a_i and the number of crossings C and twist regions n.
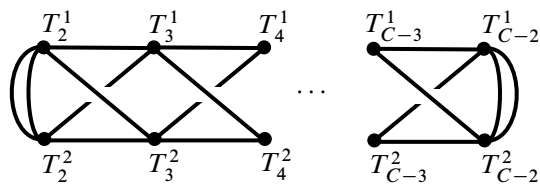

Figure 11: The form of the dual graph to a 2–bridge knot triangulation.
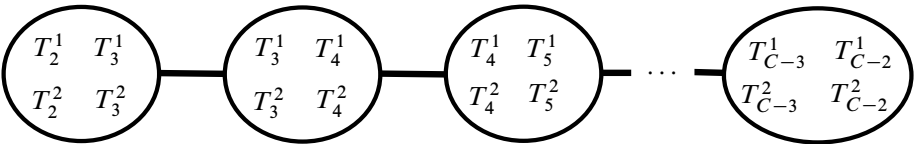
Figure 12: A tree decomposition of an ideal triangulation of a 2–bridge knot.

Finally, the above argument holds for ideal triangulations. It might be preferable to work with finite triangulations, ie tetrahedra with only finite vertices and not ideal vertices. We can modify the above decomposition into a finite triangulation by first truncating ideal vertices. This turns an ideal tetrahedron into a polyhedron with four triangular faces (from truncating) and four hexagonal faces (one for each face of the previous ideal tetrahedron). Add a finite vertex to the centre of each hexagonal face and cone to obtain six triangles. Then add a finite vertex to the centre of the polyhedron and cone to all the faces. The result is a subdivision of the ideal tetrahedron into 28 finite tetrahedra.

To obtain a tree decomposition of the finite triangulation, take the same tree decomposition as in Figure 12. Replace T_i^j with the 28 finite tetrahedra in the subdivision of T_i^j. This gives a tree decomposition, although it is likely not optimal. However, the size of each bag in the decomposition is $28 * 4 = 112$. Thus the treewidth of this decomposition is 111, which is constant, independent of C and n. $\square$

Remark 6.2 It is folklore that given a link diagram of treewidth k (seen as a 4–valent graph), one can construct an ideal triangulation of its link exterior with treewidth $O(k)$. More explicitly, this can be done using `SnapPy`'s link complement triangulation algorithm [14]. Specifically, `SnapPy`'s procedure first constructs a cell complex with four cells per crossing in the diagram, whose dual graph connects the four cells around a crossing in a square and whose arcs run along the link diagram otherwise; this graph has treewidth at most 4 times the treewidth of the diagram. The procedure concludes by contracting the cell complex along the link diagram, which induces *arc contractions* in the dual graph that can only reduce the treewidth [31], and by adding a constant number of cells (increasing the treewidth by a constant) to get an ideal triangulation.

However, the construction of Proposition 6.1 above gives a more natural triangulation with a smaller treewidth.

We have found manifolds with boundary with unbounded volume and bounded treewidth. We wish to find closed examples. To do so, we will perform Dehn filling on the 2–bridge knots from above, by attaching a layered solid torus. However, we need to

ensure that there is a triangulation of the manifold with only one vertex on the boundary, with bounded treewidth.

Theorem 6.3 *There exists a sequence of closed hyperbolic manifolds M_n with bounded treewidth and unbounded volume.*

Proof The sequence will be obtained by Dehn filling 2–bridge knot complements. By Proposition 6.1, there exists a sequence of 2–bridge knots with volume approaching infinity but with treewidth bounded by a constant.

By virtue of Corollary 3.4 (and Theorem 2.8), applying the 0–efficiency construction of Jaco and Rubinstein [20], any (compact) 2–bridge knot exterior admits a bounded triangulation with constant treewidth and one vertex on each boundary component. Now perform a very high Dehn filling on the knot. The volume decreases by a bounded amount; see for example [17]. Hence the sequence still has volume approaching infinity.

We construct the triangulation of the Dehn filling using Lemma 5.2, which increases the treewidth by one, by Lemma 3.8. Thus the treewidth remains bounded. □

Acknowledgements

References

[1]   **I Agol**, *Small 3–manifolds of large genus*, Geom. Dedicata 102 (2003) 53–64 MR

[2]   **S Arnborg**, **D G Corneil**, **A Proskurowski**, *Complexity of finding embeddings in a k–tree*, SIAM J. Algebraic Discrete Methods 8 (1987) 277–284 MR

[3]   **S Arnborg**, **A Proskurowski**, **D G Corneil**, *Forbidden minors characterization of partial 3–trees*, Discrete Math. 80 (1990) 1–19 MR

[4]   **R Benedetti**, **C Petronio**, *Lectures on hyperbolic geometry*, Springer (1992) MR

[5]   **D Bienstock**, *On embedding graphs in trees*, J. Combin. Theory Ser. B 49 (1990) 103–136 MR

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

[7]   **G Burde**, **H Zieschang**, *Knots*, De Gruyter Studies in Mathematics 5, de Gruyter, Berlin (1985) MR

[8]   **B A Burton**, *Introducing Regina, the 3–manifold topology software*, Experiment. Math. 13 (2004) 267–272 MR

[9]   **B A Burton**, *A new approach to crushing 3–manifold triangulations*, Discrete Comput. Geom. 52 (2014) 116–139 MR

[10] **B A Burton**, **R G Downey**, *Courcelle's theorem for triangulations*, J. Combin. Theory Ser. A 146 (2017) 264–294 MR

[11] **B A Burton**, **C Maria**, **J Spreer**, *Algorithms and complexity for Turaev–Viro invariants*, from "Automata, languages, and programming, I" (M M Halldórsson, K Iwama, N Kobayashi, B Speckmann, editors), Lecture Notes in Comput. Sci. 9134, Springer (2015) 281–293 MR

[12] **B A Burton**, **J Spreer**, *The complexity of detecting taut angle structures on triangulations*, from "Proceedings of the Twenty-Fourth Annual ACM–SIAM Symposium on Discrete Algorithms" (S Khanna, editor), SIAM, Philadelphia (2012) 168–183 MR

[13] **B Courcelle**, *The monadic second-order logic of graphs, I: Recognizable sets of finite graphs*, Inform. and Comput. 85 (1990) 12–75 MR

[14] **M Culler**, **N M Dunfield**, **M Goerner**, **J R Weeks**, *SnapPy: a computer program for studying the geometry and topology of 3–manifolds*, software (2016) Available at http://snappy.computop.org/

[15] **H Edelsbrunner**, *Geometry and topology for mesh generation*, Cambridge Monographs on Applied and Computational Mathematics 7, Cambridge Univ. Press (2001) MR

[16] **W Fenchel**, *Elementary geometry in hyperbolic space*, De Gruyter Studies in Mathematics 11, de Gruyter, Berlin (1989) MR

[17] **D Futer**, **E Kalfagianni**, **J S Purcell**, *Dehn filling, volume, and the Jones polynomial*, J. Differential Geom. 78 (2008) 429–464 MR

[18] **F Guéritaud**, *On canonical triangulations of once-punctured torus bundles and two-bridge link complements*, Geom. Topol. 10 (2006) 1239–1284 MR With an appendix by David Futer

[19] **K Huszár**, **J Spreer**, **U Wagner**, *On the treewidth of triangulated 3–manifolds*, from "34th International Symposium on Computational Geometry" (B Speckmann, C D Tóth, editors), Leibniz Int. Proc. Inform. 99, Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern (2018) art. id. 46 MR

[20] **W Jaco**, **J H Rubinstein**, *0–efficient triangulations of 3–manifolds*, J. Differential Geom. 65 (2003) 61–168 MR

[21] **W Jaco**, **J H Rubinstein**, *Layered-triangulations of 3–manifolds*, preprint (2006) arXiv

[22] **W Jaco**, **E Sedgwick**, *Decision problems in the space of Dehn fillings*, Topology 42 (2003) 845–906 MR

[23] **D A Každan**, **G A Margulis**, *A proof of Selberg's hypothesis*, Mat. Sb. 75 (117) (1968) 163–168 MR In Russian; translated in Math. USSR-Sb. 4 (1968) 147–152

[24] **B Kleiner**, **J Lott**, *Notes on Perelman's papers*, Geom. Topol. 12 (2008) 2587–2855 MR

[25] **T Kobayashi**, **Y Rieck**, *A linear bound on the tetrahedral number of manifolds of bounded volume (after Jørgensen and Thurston)*, from "Topology and geometry in dimension three" (W Li, L Bartolini, J Johnson, F Luo, R Myers, J H Rubinstein, editors), Contemp. Math. 560, Amer. Math. Soc., Providence, RI (2011) 27–42 MR

[26] **C Maria**, **J Spreer**, *A polynomial time algorithm to compute quantum invariants of 3–manifolds with bounded first Betti number*, from "Proceedings of the Twenty-Eighth Annual ACM–SIAM Symposium on Discrete Algorithms" (P N Klein, editor), SIAM, Philadelphia (2017) 2721–2732 MR

[27] **G D Mostow**, *Strong rigidity of locally symmetric spaces*, Annals of Mathematics Studies 78, Princeton Univ. Press (1973) MR

[28] **G Perelman**, *The entropy formula for the Ricci flow and its geometric applications*, preprint (2002) arXiv

[29] **G Perelman**, *Ricci flow with surgery on three-manifolds*, preprint (2003) arXiv

[30] **J S Purcell**, *Hyperbolic knot theory*, book project (2018) Available at http://users.monash.edu/~jpurcell/hypknottheory.html

[31] **N Robertson**, **P D Seymour**, *Graph minors, II: Algorithmic aspects of tree-width*, J. Algorithms 7 (1986) 309–322 MR

[32] **M Sakuma**, **J Weeks**, *Examples of canonical decompositions of hyperbolic link complements*, Japan. J. Math. 21 (1995) 393–439 MR

[33] **A Satyanarayana**, **L Tung**, *A characterization of partial 3–trees*, Networks 20 (1990) 299–322 MR

[34] **P D Seymour**, **R Thomas**, *Call routing and the ratcatcher*, Combinatorica 14 (1994) 217–241 MR

[35] **D M Thilikos**, **M J Serna**, **H L Bodlaender**, *Constructive linear time algorithms for small cutwidth and carving-width*, from "Algorithms and computation" (D T Lee, S-H Teng, editors), Lecture Notes in Comput. Sci. 1969, Springer (2000) 192–203 MR

[36] **W P Thurston**, *The geometry and topology of three-manifolds*, lecture notes, Princeton University (1979) Available at http://msri.org/publications/books/gt3m

INRIA Sophia Antipolis-Méditerranée
Valbonne, France

School of Mathematics, Monash University
Monash University, VIC, Australia

clement.maria@inria.fr, jessica.purcell@monash.edu