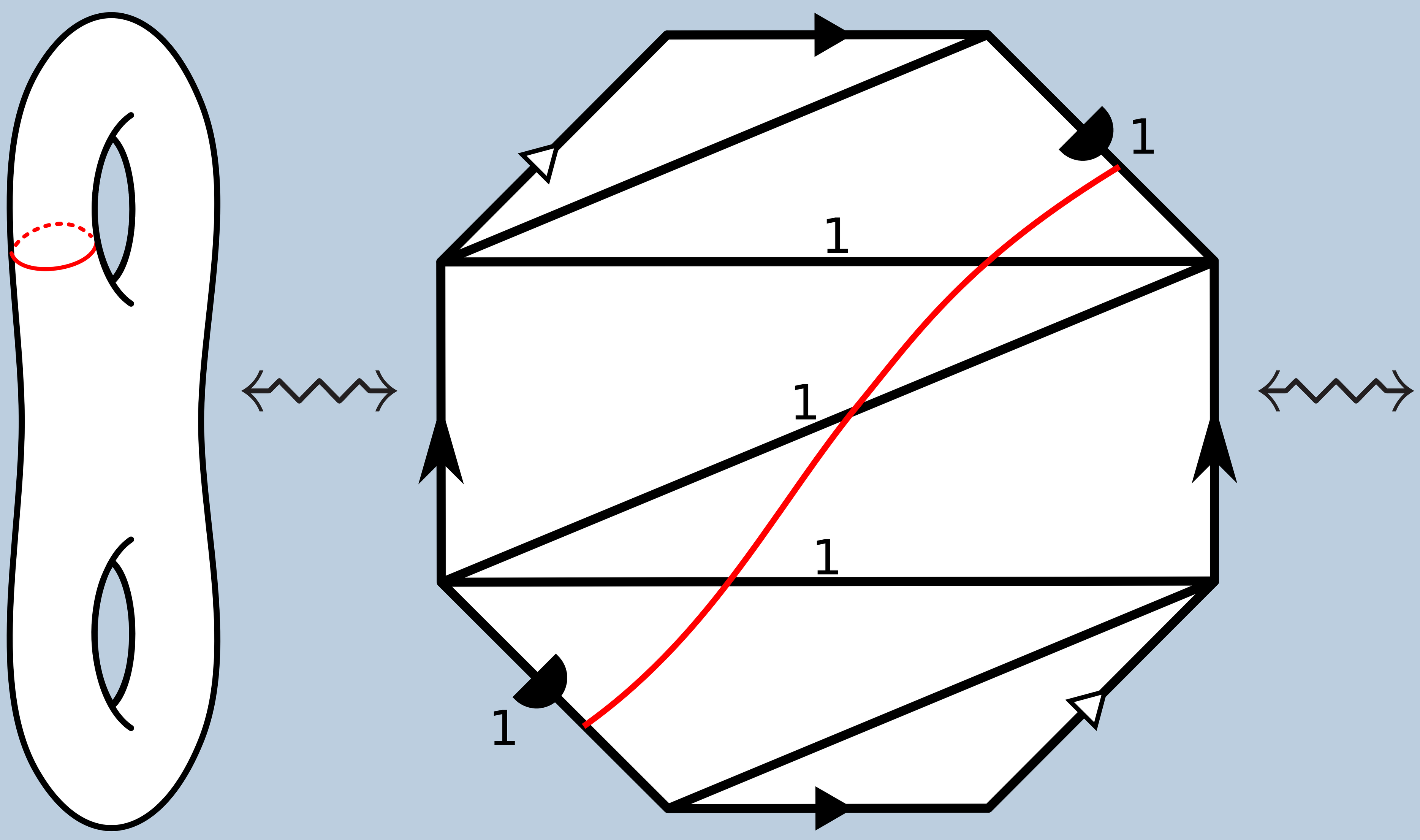
How many loops are on this surface?

Mark Bell
Mathematics Institute, University of Warwick
m.c.bell@warwick.ac.uk

THE UNIVERSITY OF WARWICK

Problem

A collection of curves Γ on a surface S can be described by the number of times it meets each edge of a triangulation $\mathcal{T}$ [4]. If $\mathcal{T}$ has edges $e_1, \ldots, e_k$ let $\underline{v} = \underline{v}(\Gamma) := (\iota(\Gamma, e_1) \cdots \iota(\Gamma, e_k))$.



$$\underline{v} := \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \end{pmatrix}$$

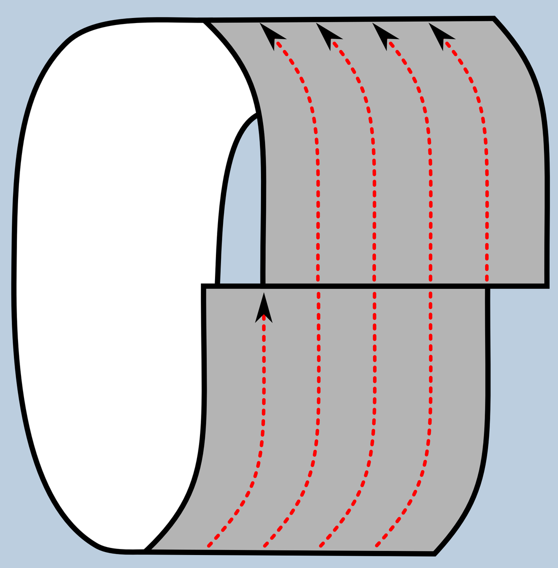
Given $\underline{v}$ decide how many components Γ has.

The Street Algorithm [3]

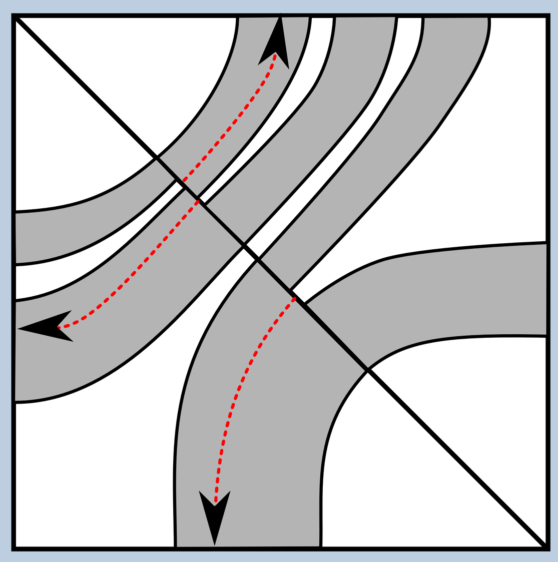
Key observation: If $|\underline{v}|$ is very large then there are many arcs in each triangle and therefore there must be many long sections where the curves run parallel.

We keep track of these parallel regions, each of which is either a rectangle or an annulus. We refer to these as *open* and *closed streets* respectively. In fact there are only ever at most $2k$ streets.

```
Start with an open street for each type of arc in each triangle
while there is an open street do
    if a street meets itself exactly then
        replace it with a closed street
    else if a street meets itself then
        replace it with a spiral
    else
        merge all open streets
    end if
end while
```



Spiraling misaligned streets



Merging open streets

At the end, each street is an annulus and the number of components of Γ is the sum of their widths. However, after the outer loop is repeated a constant number of times the width of the widest open street drops by a definite fraction.

Hence, this only requires $O(\log(|\underline{v}|))$ time and space.

Uses

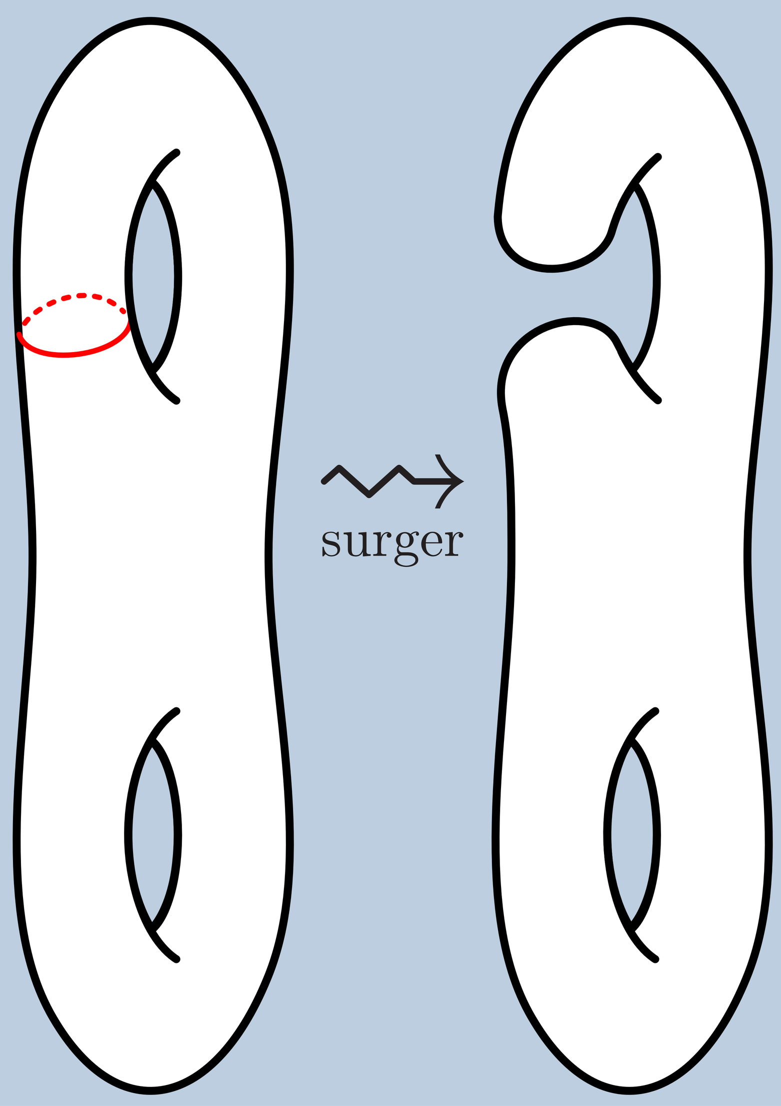
We can *surger* S along Γ to obtain a new surface S_Γ. If $\mathcal{T}$ is a triangulation of S then it can also be surgered, giving a triangulation $\mathcal{T}_\Gamma$ of S_Γ.

Lemma (Bell). *We can compute $\mathcal{T}_\Gamma$ from $\mathcal{T}$ in $O(\log(|\underline{v}(\Gamma)|))$ time and space.*

This is a very useful technique as it allows us to construct arguments based on induction on the genus of S. For example:

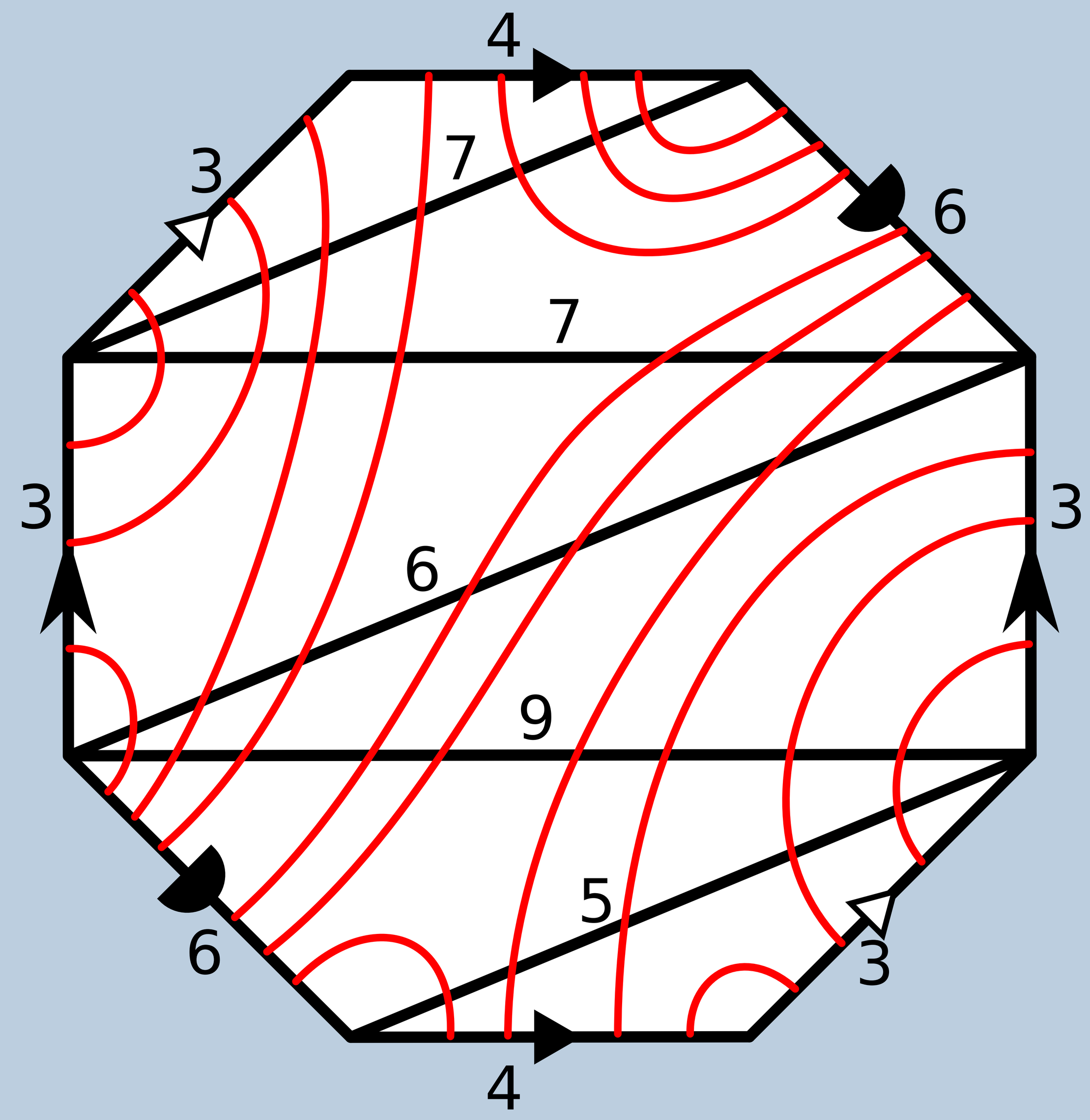
Theorem (Bell [2]). *Suppose that H is a finite set of homeomorphisms of S. There is a computable constant K such that for each $h \in \langle H \rangle$ there is a collection of curves Γ fixed by h (up to isotopy) such that the homeomorphism induced by h on S_Γ fixes no curves and*

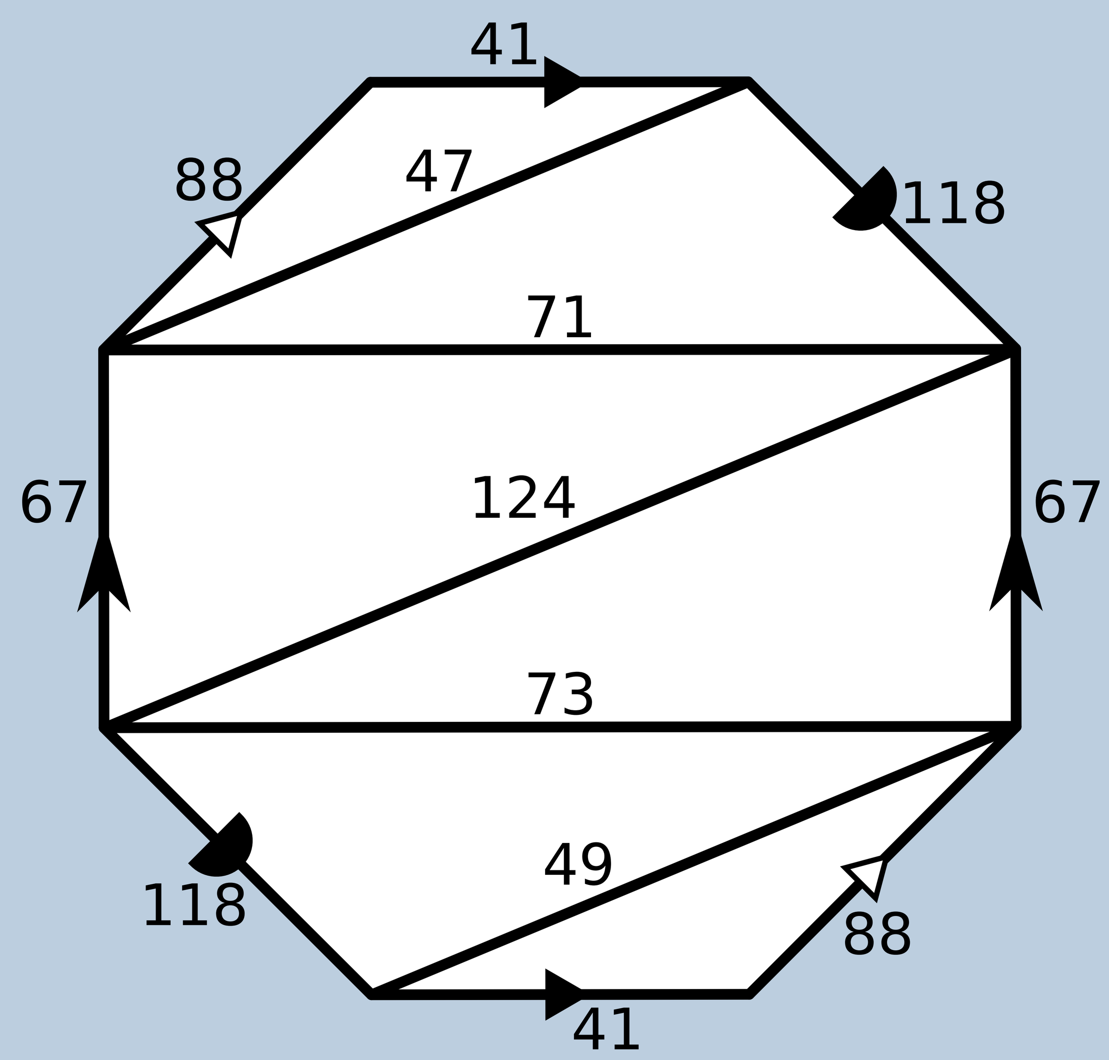
$$\log(|\underline{v}(\Gamma)|) \leq K\ell(h).$$



Examples

In each case: how many curves are there? How did you figure this out?





You can create your own examples by choosing numbers that obey the *even* triangle inequality.

Naïve Algorithm

```
while there is an unmarked arc do
    Get an unmarked arc
    repeat
        Mark the current arc
        Follow along to next arc
    until back at the starting point
end while
```

The number of components of Γ is the number of times the outer loop is repeated. However, this involves crossing every arc in every triangle.

Hence, this requires $O(|\underline{v}|)$ time and space.

References

[1] Ian Agol, Joel Hass and William Thurston: *The computational complexity of knot genus and spanning area*, Transactions of the American Mathematical Society, (2006)

[2] Mark C. Bell: *An algorithm for deciding reducibility*, ArXiv e-prints: 1403.2997 (2014)

[3] Jeff Erickson and Amir Nayyeri: *Tracing compressed curves in triangulated surfaces*, Discrete & Computational Geometry. (2013)

[4] Marcus Schaefer, Eric Sedgwick and Daniel Štefankovič: *Algorithms for normal curves and surfaces*, Proceedings of 8th International Conference on Computing and Combinatorics (2002)