Knot diagram recovery in exponential time

R. Haraway, N. Hoffman, S. Schleimer, E. Sedgwick

AMS 2020 Fall SE Sectional Meeting

The problem

The problem

```
from snappy import *
LX = LinkExteriors
HX = HTLinkExteriors
unkn = lambda M: not(LX.identify(M) or HX.identify(M))
first = lambda itr: itr.__next__()
census_entry = lambda manifold: manifold.identify()[0]
M = census_entry(first(filter(unkn, CensusKnots))); M
>> m082(0,0)
M.num_tetrahedra()
>> 4
```

Previous work

Construction (ABGKLMOSTWW '19)

Diagrams for the knot exteriors t12533, t12681, o9_38928, o9_39162, o9_40363, o9_40487, o9_40504, o9_40582, and o9_42675.

Previous work

Construction (ABGKLMOSTWW '19)

Diagrams for the knot exteriors t12533, t12681, o9_38928, o9_39162, o9_40363, o9_40487, o9_40504, o9_40582, and o9_42675.

The biggest diagram, for o9_40363, had 72 crossings.

Previous work

Knot diagram recovery is ***not*** in PSPACE!

Previous work

Knot diagram recovery is ***not*** in PSPACE!

Fact:

Let $F_0 = 1, F_1 = 2, F_{n+2} = F_{n+1} + F_n$ *be the Fibonacci numbers. For* $n \geq 1$, *the* (F_n, F_{n+1})*-torus knot* Fib_n *is an edge in a triangulation of* S^3 *with* $2n - 1$ *tetrahedra. But* $cr(Fib_n) \geq \phi^{n+1}$.

Previous work

Knot diagram recovery is in EXPSPACE.

Previous work

Knot diagram recovery is in EXPSPACE.

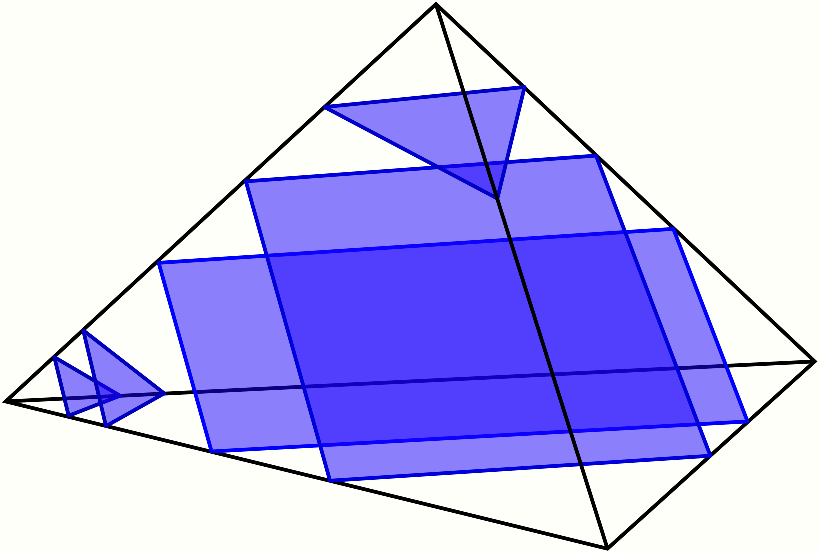
Theorem (King 2001)

Let $\mathcal{T}$ be a triangulation of S^3 with n tetrahedra, and let $L \subset \mathcal{T}^1$ be a link. [Then] $cr(L) < 2^{810n^2}$.

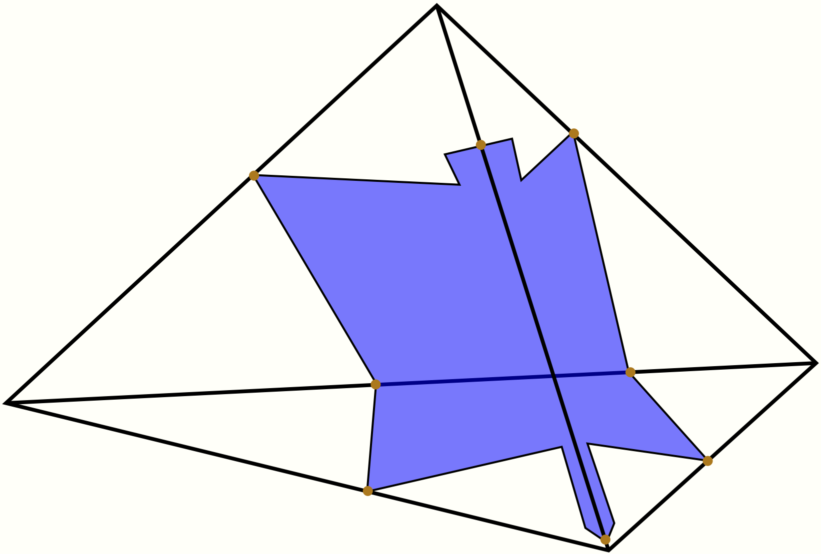
Theorem (HHSS '20)

There is a polynomial p and an algorithm that, given a link subcomplex K of a triangulation $\mathcal{T}$ of S^3, will construct a diagram of K in at most $2^{p(|\mathcal{T}|)}$ seconds.

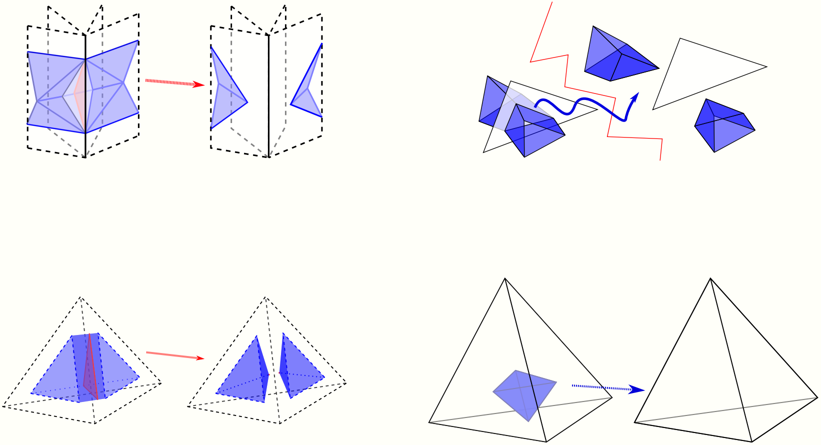
Normal Surfaces



Almost Normal* Surfaces



Normalizing Surfaces in 3D



How To Recognize* S^3

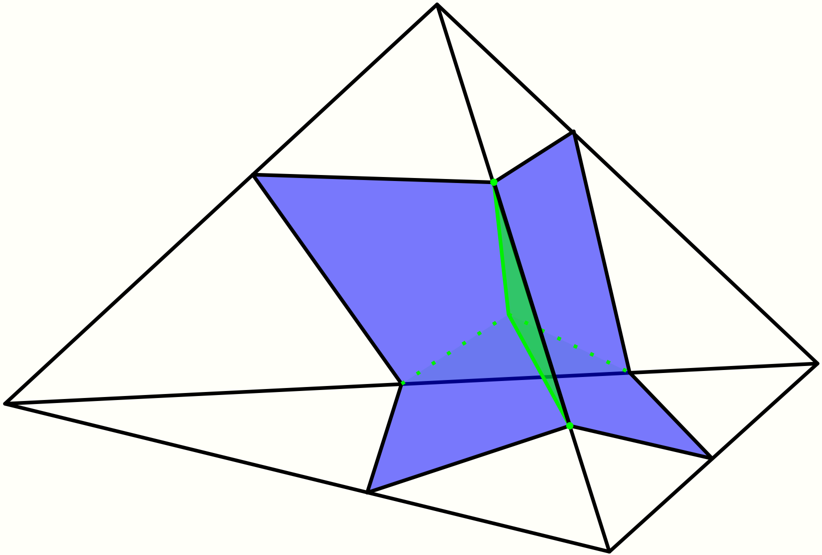
If an almost-normal sphere normalizes to a trivial surface on both sides, then the triangulation is S^3.

Algorithm (King, HHSS)

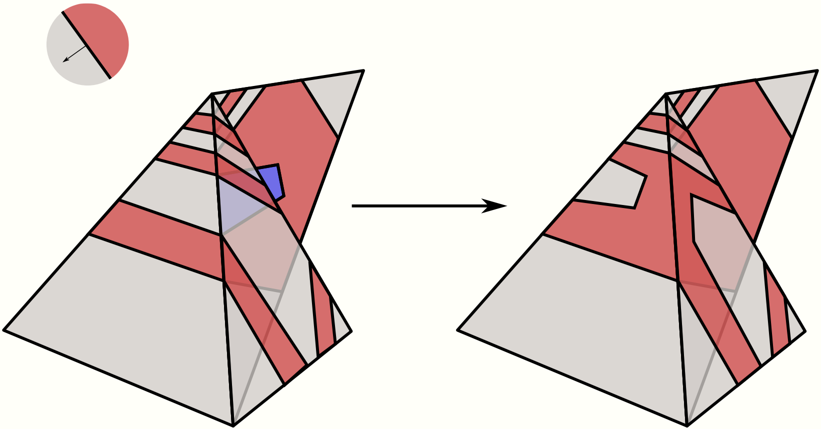
Given a triangulation $\mathcal{T}$ *of* S^3, *in time exponential in* $|\mathcal{T}|$, *construct an almost-normal** *bridge sphere.*

This is implicit in King's work; we improve the bounds and make the algorithm explicit.

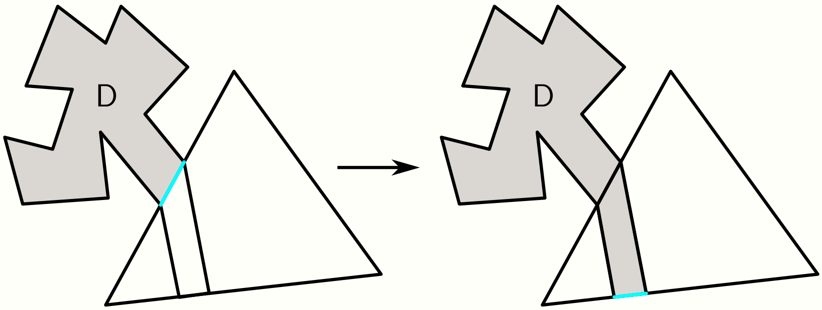
Exceptional Bridge Discs



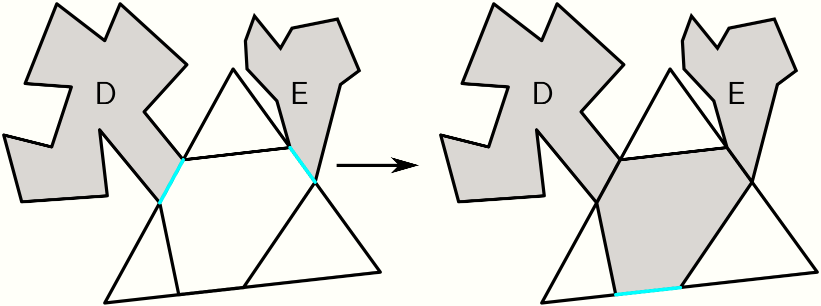
Normalizing Surfaces using $\mathcal{T}^{(2)}$



Other Bridge Discs: Trapezoids



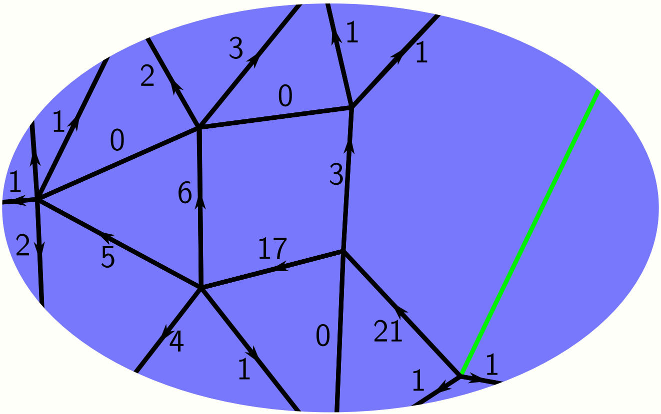
Other Bridge Discs: Hexagons



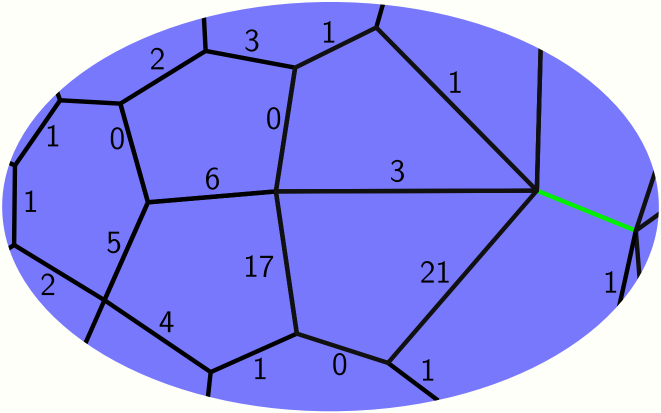
A bridge disc
is determined by its counts
of trapezoids and hexagons.

A bridge disc
is determined by its intersection
with the bridge sphere.

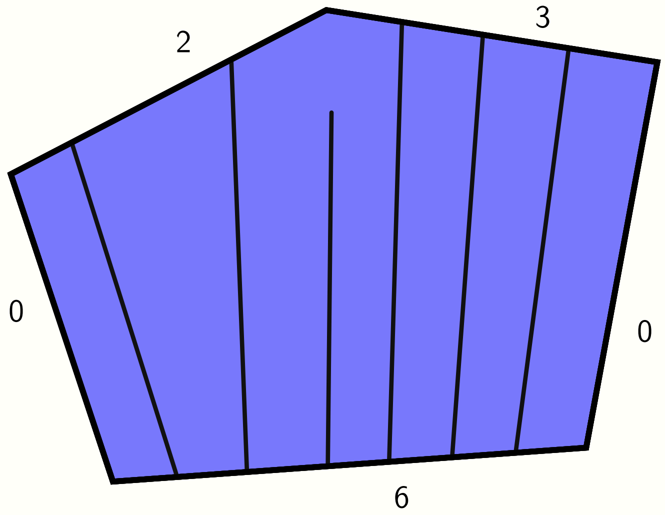
Bridge Arcs by Coordinates



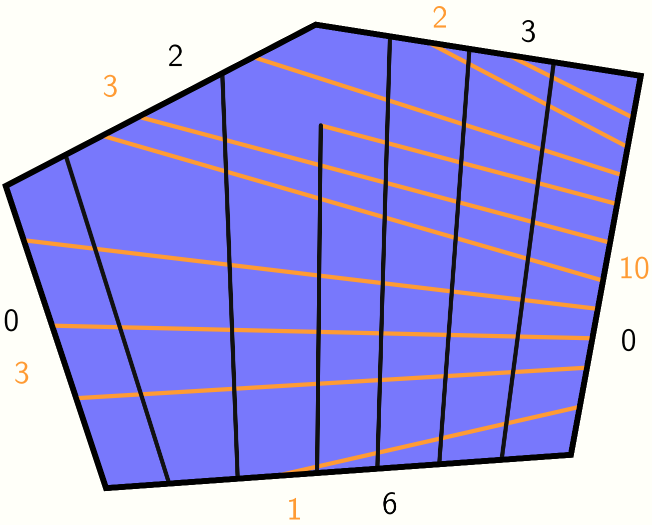
Dualize the Normal Surface



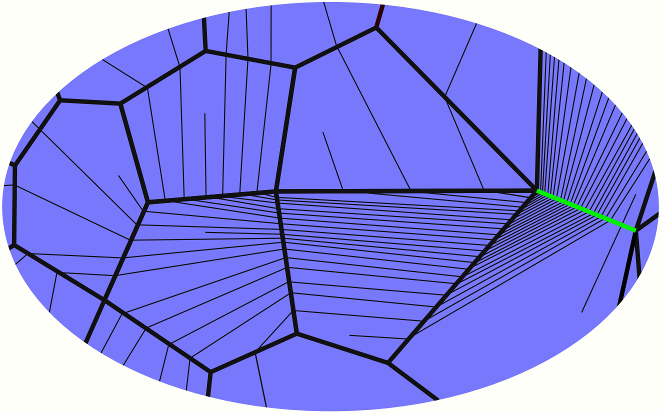
The Diagram at a Dual Cell: Upper



The Diagram at a Dual Cell



Draw Diagrams in all the Dual Cells



Sketch of Proof of Main Theorem: 0

We can compute an almost-normal* bridge sphere S in EXPTIME; compute S.

S has at most exponentially many discs.

Sketch of Proof: 1

There are also at most exponentially many complementary trapezoids and hexagons.

Calculating the associated bridge arc on S given in terms of coordinates takes at most EXPTIME per bridge.

Since there are only exponentially many bridges, calculating all the bridge arcs' coordinates can be done in EXPTIME.

Sketch of Proof: 2

The local picture of the upper (or lower) bridge arcs at a dual cell is determined by:

- the coordinates at the edges, and
- the location of the ending-arc's boundary point on the cell's boundary (if the cell has an ending-arc).

Sketch of Proof: 2

Going through each cell in turn, ensuring that ending-arcs are connected to the appropriate points, requires at most an exponential amount of auxiliary space, and an exponential amount of time per cell. That's an exponential amount of time per side. □

Where To From Here

- Improve on King's bounds.
- Design an algorithm to work natively with knot exteriors instead of with triangulations of S^3.
- Implement an algorithm and run it on the 1069 knot exteriors without known diagrams.