Warning. The second formula on page (-2) "**Conclusion**" is silly-wrong. A fix will be posted here soon: some of the numbers written in this handout are a bit off, yet the qualitative results remain exactly the same (namely, for finite type, 3D seems to beat 2D, with the same algorithms).

Yarn-Ball Knots

[K-OS] on October 21, 2021

Dror Bar-Natan with Itai Bar-Natan, Iva Halacheva, and Nancy Scherich

Agenda. A modest light conversation on how knots should be measured.

Abstract. Let there be scones! Our view of knot theory is biased in favour of pancakes.
Technically, if K is a 3D knot that fits in volume V (assuming fixed-width yarn), then its projection to 2D will have about $V^{4/3}$ crossings. You'd expect genuinely 3D quantities associated with K to be computable straight from a 3D presentation of K. Yet we can hardly ever circumvent this $V^{4/3} \gg V$ "projection fee". Exceptions include linking numbers (as we shall prove), the hyperbolic volume, and likely finite type invariants (as we shall discuss in detail). But knot polynomials and knot homologies seem to always pay the fee. Can we exempt them?

More at http://drorbn.net/kos21

Thanks for inviting me to speak at [K-OS]!

Most important: http://drorbn.net/kos21

See also arXiv:2108.10923.

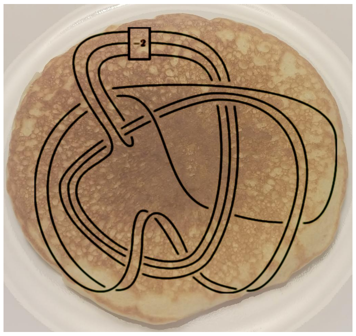
If you can, please turn your video on! (And mic, whenever needed).

A recurring question in knot theory is "do we have a 3D understanding of our invariant?"

- See Witten and the Jones polynomial.
- See Khovanov homology.

I'll talk about my perspective on the matter…

We often think of knots as planar diagrams. 3-dimensionally, they are embedded in "pancakes".

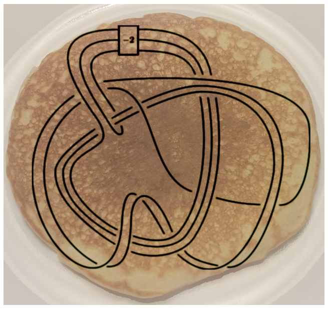


Knot by Lisa Piccirillo, pancake by DBN

But real life knots are 3D!

A Yarn Ball

'Connector' by Alexandra Griess and Jorel Heid (Hamburg, Germany). Image from www.waterfrontbia.com/ice-breakers-2019-presented-by-ports/.



The difference matters when

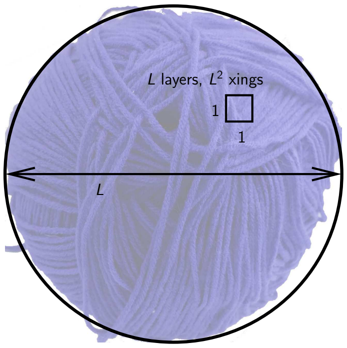
- We make statements about "random knots".
- We figure out computational complexity.

Let's try to make it quantitative…

$V \sim L^3$

$n = $ xing number $\sim L^2L^2 = L^4 = V^{4/3}$

("$\sim$" means "equal up to constant terms and log terms")

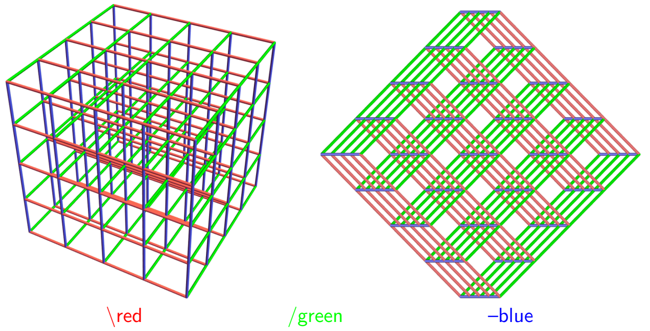


Conversation Starter 1. A knot invariant ζ is said to be Computationally 3D, or C3D, if

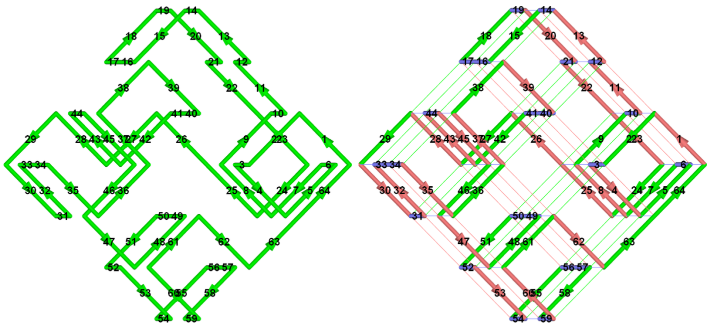
$$C_\zeta(3D, V) \ll C_\zeta(2D, V^{4/3}).$$

This isn't a rigorous definition! It is time- and naïveté-dependent! But there's room for less-stringent rigour in mathematics, and on a philosophical level, our definition means something.

Theorem 1. Let lk denote the linking number of a 2-component link. Then $C_{lk}(2D, n) \sim n$ while $C_{lk}(3D, V) \sim V$, so lk is C3D!
Proof. WLOG, we are looking at a link in a grid, which we project as on the right:

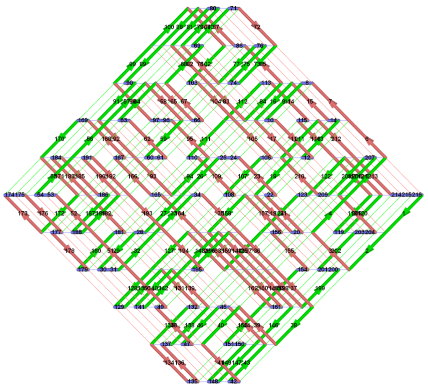


Here's what it look like, in the case of a knot:



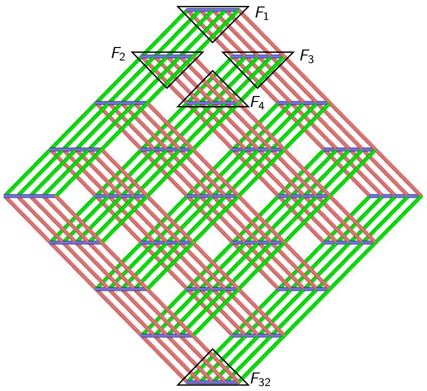
And here's a bigger knot.

This may look like a lot of information, but if V is big, it's less than the information in a planar diagram, and it is easily computable.



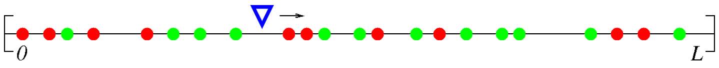
There are $2L^2$ triangular "crossings fields" F_k in such a projection.

WLOG, in each F_k all over strands and all under strands are oriented in the same way and all green edges belong to one component and all red edges to the other.

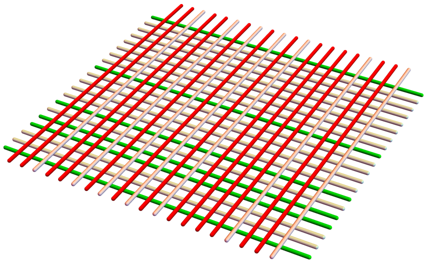


So $2L^2$ times we have to solve the problem "given two sets R and G of integers in $[0, L]$, how many pairs $\{(r, g) \in R \times G\colon r < g\}$ are there?". This takes time $\sim L$ (see below), so the overall computation takes time $\sim L^3$.

Below. Start with $rb = cf = 0$ ("reds before" and "cases found") and slide ▽ from left to right, incrementing rb by one each time you cross a ● and incrementing cf by rb each time you cross a ●:



In general, with our limited tools, speedup arises because appropriately projected 3D knots have many uniform "red over green" regions:



Great Embarrassment 1. I don't know if any of the Alexander, Jones, HOMFLY-PT, and Kauffman polynomials is C3D. I don't know if any Reshetikhin-Turaev invariant is C3D. I don't know if any knot homology is C3D.

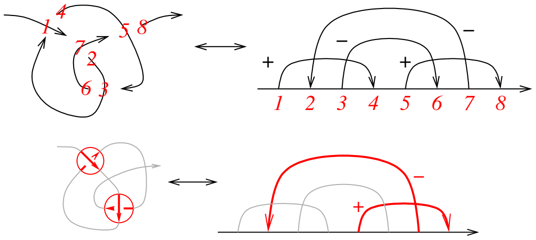
Or maybe it's a cause for optimism — there's still something very basic we don't know about (say) the Jones polynomial. Can we understand it well enough 3-dimensionally to compute it well? If not, why not?

Next we argue that most finite type invariants are probably C3D...

(What a weak statement!)

All pre-categorification knot polynomials are power series whose coefficients are finite type invariants. (This is sometimes helpful for the computation of finite type invariants, but rarely helpful for the computation of knot polynomials).

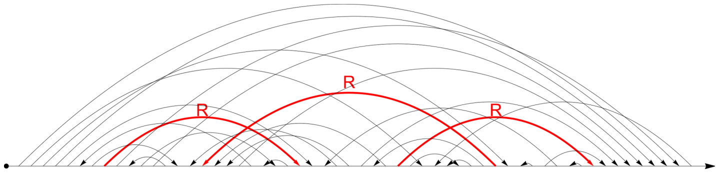
Gauss diagrams and sub-Gauss-diagrams:



Let $\varphi_d\colon \{\text{knot diagrams}\} \to \langle\text{Gauss diagrams}\rangle$ map every knot diagram to the sum of all the sub-diagrams of its Gauss diagram which have at most d arrows.

Under-Explained Theorem (Goussarov-Polyak-Viro). A knot invariant ζ is of type d iff there is a linear functional ω on $\langle\text{Gauss diagrams}\rangle$ such that $\zeta = \omega \circ \varphi_d$.

Proof of Theorem FT2D.



We need to count how many times a diagram such as the red appears within a bigger diagram, having n arrows. Clearly this can be done in time $\sim n^3$, and in general, in time $\sim n^d$.

Conversation Starter 2. Similarly, if η is a stingy quantity (a quantity we expect to be small for small knots), we will say that η has Savings in 3D, or "has S3D" if $M_\eta(3D, V) \ll M_\eta(2D, V^{4/3})$.

Example (R. van der Veen, D. Thurston, private communications). The hyperbolic volume has $S3D$.

Great Embarrassment 2. I don't know if the genus of a knot has S3D! In other words, even if a knot is given in a 3-dimensional, the best way I know to find a Seifert surface for it is to first project it to 2D, at a great cost.

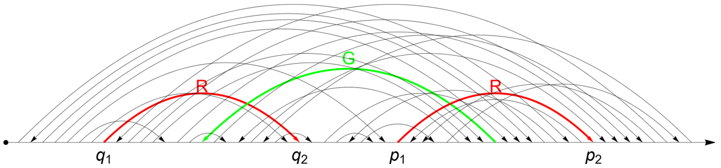
Theorem FT2D. If ζ is a finite type invariant of type d then $C_\zeta(2D, n)$ is at most $\sim n^{\lfloor 3d/4 \rfloor}$. With more effort, $C_\zeta(2D, n) \lesssim n^{(\frac{2}{3}+\epsilon)d}$.

Note that there are some exceptional finite type invariants, e.g. high coefficients of the Alexander polynomial and other poly-time knot polynomials, which can be computed much faster!

Theorem FT3D. If ζ is a finite type invariant of type d then $C_\zeta(3D, V)$ is at most $\sim V^{6d/7+1/7}$. With more effort, $C_\zeta(2D, V) \lesssim V^{(\frac{4}{5}+\epsilon)d}$.

Tentative Conclusion. As
$n^{3d/4} \sim (V^{4/3})^{3d/4} = V \gg V^{6d/7+1/7}$ $\qquad n^{2d/3} \sim (V^{4/3})^{2d/3} = V^{8d/9} \gg V^{4d/5}$
these theorems say "most finite type invariants are probably C3D; the ones in greater doubt are the lucky few that can be computed unusually quickly".

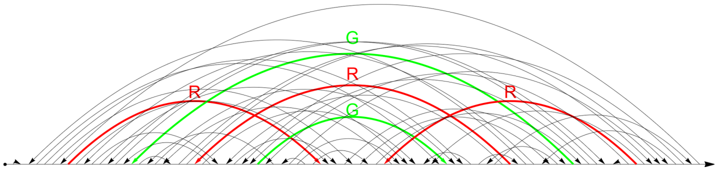
Theorem FT2D. If ζ is a finite type invariant of type d then $C_\zeta(2D, n)$ is at most $\sim n^{\lfloor 3d/4 \rfloor}$. With more effort, $C_\zeta(2D, n) \lesssim n^{(\frac{2}{3}+\epsilon)d}$.



With an appropriate look-up table, it can also be done in time $\sim n^2$ (in general, $\sim n^{d-1}$). That look-up table $(T^{p_1,p_2}_{q_1,q_2})$ is of size (and production cost) $\sim n^4$ if you are naive, and $\sim n^2$ if you are just a bit smarter. Indeed

$$T^{p_1,p_2}_{q_1,q_2} = T^{0,p_2}_{0,q_2} - T^{0,p_1}_{0,q_2} - T^{0,p_2}_{0,q_1} + T^{0,p_1}_{0,q_1},$$

and $(T^{0,p}_{0,q})$ is easy to compute.



With multiple uses of the same lookup table, what naively takes $\sim n^5$ can be reduced to $\sim n^3$.

In general within a big d-arrow diagram we need to find an as-large-as possible collection of arrows to delay. These must be non-adjacent to each other. As the adjacency graph for the arrows is at worst quadrivalent, we can always find $\lceil\frac{d}{4}\rceil$ non-adjacent arrows, and hence solve the counting problem in time $\sim n^{d-\lceil\frac{d}{4}\rceil} = n^{\lfloor 3d/4\rfloor}$.

Note that this counting argument works equally well if each of the d arrows is pulled from a different set!

It follows that we can compute φ_d in time $\sim n^{\lfloor 3d/4\rfloor}$.

□

With bigger look-up tables that allow looking up "clusters" of G arrows, we can reduce this to $\sim n^{(\frac{2}{3}+\epsilon)d}$.

□

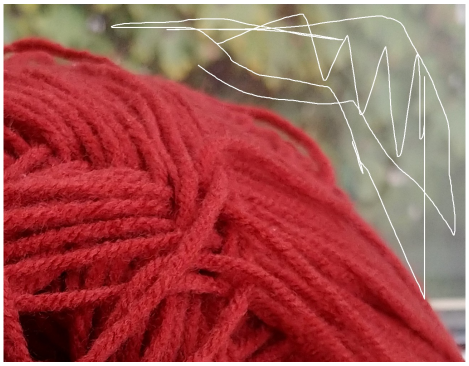
On to

Theorem FT3D. If ζ is a finite type invariant of type d then $C_\zeta(3D,V)$ is at most $\sim V^{6d/7+1/7}$. With more effort, $C_\zeta(2D,V)\lesssim V^{(\frac{4}{5}+\epsilon)d}$.

An image editing problem:

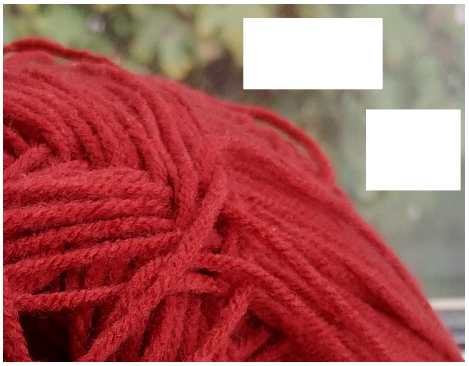
(Yarn ball and background coutesy of Heather Young)

The line/feather method:



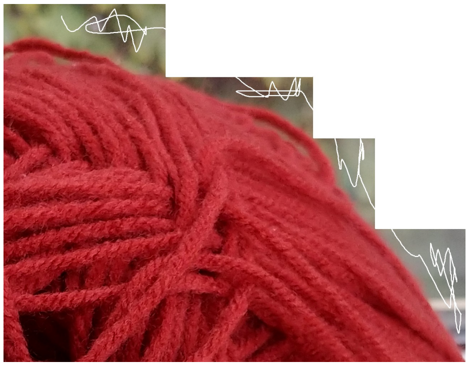
Accurate but takes forever.

The rectangle/shark method:



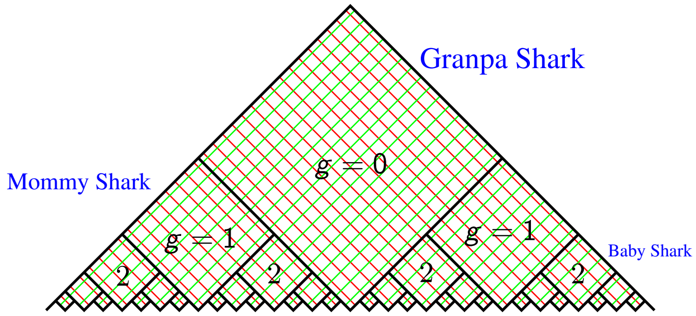
Coarse but fast.

In reality, you take a few shark bites and feather the rest ...



... and then there's an optimization problem to solve: when to stop biting and start feathering.

The structure of a crossing field.



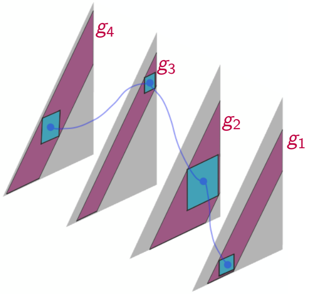
There are about $\log_2 L$ "generations". There are 2^g bites in generation g, and the total number of crossings in them is $\sim L^2/2^g$.
Let's go hunt!

Multi-feathers and multi-sharks.

For a type d invariant we need to count d-tuples of crossings, and each has its own "generation" g_i. So we have the "multi-generation"

$$\bar{g} = (g_1, \dots, g_d).$$

Let $G := \sum g_i$ be the "overall generation". We will choose between a "multi-feather" method and a "multi-shark" method based on the size of G.



The effort to take a single multi-bite is tiny. Indeed,

Lemma Given $2d$ finite sets $B_i = \{t_{i1}, t_{i2}, \dots\} \subset [1..L^3]$ and a permutation $\pi \in S_{2n}$ the quantity

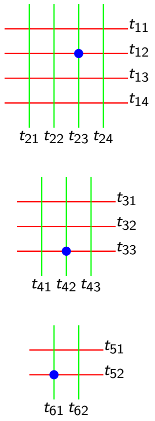
$$N = \left|\left\{ (b_i) \in \prod_{i=1}^{2d} B_i \colon \text{the } b_i\text{'s are ordered as } \pi \right\}\right|$$

can be computed in time $\sim \sum |B_i| \sim \max |B_i|$.

Proof. WLOG $\pi = Id$. For $\iota \in [1..2d]$ and $\beta \in B := \cup B_i$ let

$$N_{\iota,\beta} = \left|\left\{ (b_i) \in \prod_{i=1}^{\iota} B_i \colon b_1 < b_2 < \ldots < b_\iota \leq \beta \right\}\right|.$$

We need to know $N_{2d,\max B}$; compute it inductively using $N_{\iota,\beta} = N_{\iota,\beta'} + N_{\iota-1,\beta'}$, where β' is the predecessor of β in B. □



Conclusion. We wish to compute the contribution to φ_d coming from d-tuples of crossings of multi-generation $\bar{g}$.

- The multi-shark method does it in time

$$\sim (\text{no. of bites}) \cdot (\text{time per bite}) = L^{2d} 2^G \cdot \frac{L}{2^{\min \bar{g}}} < L^{2d+1} 2^G$$

(increases with G).

- The multi-feather method (project and use the 2D algorithm) does it in time

$$\sim (\text{no. of crossings})^{\lfloor \frac{3}{4} d \rfloor} = \left(\prod_{i=1}^{d} L^2 \frac{L^2}{2^{g_i}} \right)^{\lfloor \frac{3}{4} d \rfloor} < \frac{L^{3d}}{(2^G)^{3/4}}$$

(decreases with G).

Of course, for any specific G we are free to choose whichever is better, shark or feather.

The two methods agree (and therefore are at their worst) if $2^G = L^{\frac{4}{7}(d-1)}$, and in that case, they both take time $\sim L^{\frac{18}{7}d + \frac{3}{7}} = V^{\frac{6}{7}d + \frac{1}{7}}$.

The same reasoning, with the $n^{(\frac{2}{3}+\epsilon)d}$ feather, gives $V^{(\frac{4}{5}+\epsilon)d}$.

□

If time — a word about braids.

Thank You!