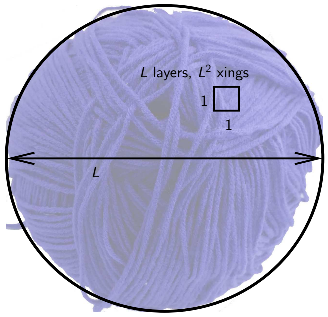
$V \sim L^3$

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

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

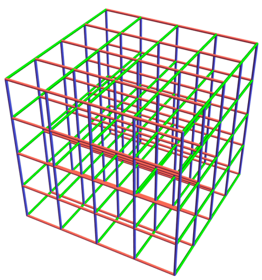


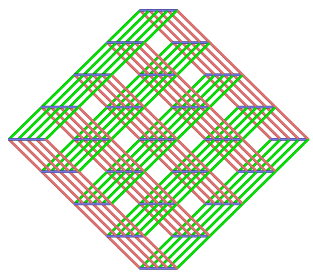
Conversation Starter 1. A knot invariant ζ is said to be Computationally 3D, or C3D, if

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

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.

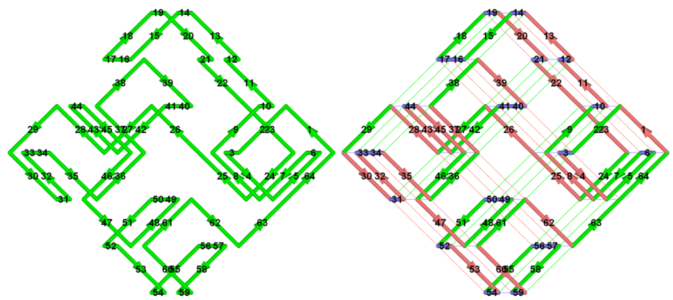
Theorem 1. Let lk denote the linking number of a 2-component link. Then $C_{lk}(2D, n) \sim n$ while $C_{lk}(3D, V) \sim V$, so lk is C3D!
Proof. WLOG, we are looking at a link in a grid, which we project as on the right:





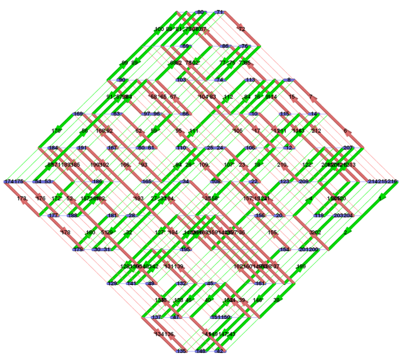
\red /green –blue

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



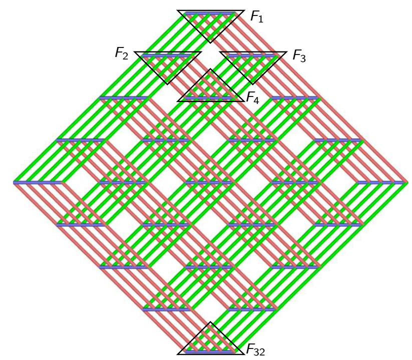
And here's a bigger knot.

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.



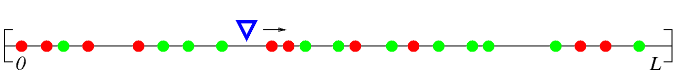
There are $2L^2$ triangular "crossings fields" F_k in such a projection.

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.

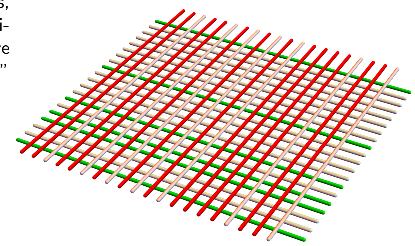


So $2L^2$ times we have to solve the problem "given two sets R and G of integers in $[0, L]$, how many pairs $\{(r, g) \in R \times G : r < g\}$ are there?". This takes time $\sim L$ (see below), so the overall computation takes time $\sim L^3$.

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



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



Video and more at http://www.math.toronto.edu/~drorbn/Talks/KOS-211021/