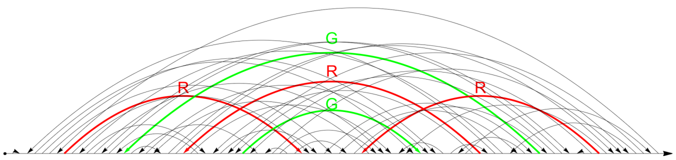


With multiple uses of the same lookup table, what naively takes $\sim n^5$ can be reduced to $\sim n^3$.

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 $\sim n^{d-\lceil \frac{d}{4} \rceil} = n^{\lfloor 3d/4 \rfloor}$.

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, $C_\zeta(2D, V) \lesssim V^{(\frac{4}{5}+\epsilon)d}$.

The line/feather method:

Accurate but takes forever.

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.

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 $\sim n^{\lfloor 3d/4 \rfloor}$.

$\square$

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

$\square$

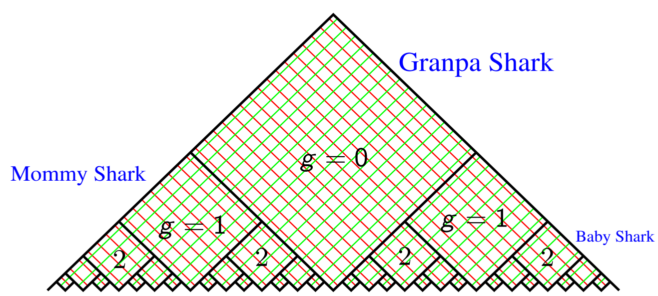
An image editing problem:

(Yarn ball and background coutesy of Heather Young)

The rectangle/shark method:

Coarse but fast.

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!

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