Editor's comment: Based on the advice received, I have decided that your manuscript could be reconsidered for publication should you be prepared to incorporate major revisions. When preparing your revised manuscript, you are asked to carefully consider the reviewer comments which can be found below, and submit a list of responses to the comments. You are kindly requested to also check the website for possible reviewer attachment(s).
We look forward to receiving your revised manuscript before 19 Oct 2022.

Reviewer #1: There are two common equivalent ways to represent knots: either by a planar diagram or by a 3-dimensional path, say in the grid Z^3 to make it finite. Do these two representations lead to different behaviors of knot invariants? In what sense? These questions were studied in the literature of biological DNA knotting, and models of random knotting. They seem to be a good starting point to advance our understanding of knot theory. The writers of this paper also ask this question, and suggest approaching it using computational complexity. They show that certain invariants that are standardly defined and computed using 2D knot diagrams seem to be easier to compute directly using their 3D representation.
Theorem 2.1 gives a "3D" algorithm to compute the linking number. The authors reduce it to a combinatorial problem of counting all pairs $a_i < b_j$ given two lists $\{a_i\}$ and $\{b_j\}$, which can be done in $O(n \log n)$, instead of $O(n^2)$, if one sorts the two lists. The reduction is elegant but also fairly standard. Theorem 3.3 gives an algorithm for computing a finite type invariant of order d, in n^d time up to log n factors. In the summation over d crossings, they apply d divide-and-conquers to count each participating crossing, again improving each naïve $O(n^2)$ to $O(n \log n)$.
The main definitions and statements in pages 1-5 are vague and imprecise, and do not meet the general standard of rigor in mathematical communication. The main notion of "computationally 3D" is conditional on our current knowledge of algorithms (see page 2 conversation starter 1 and the following paragraph). This contrasts with the common practice in computational complexity, where definitions are unambiguous and clearly stated, and then results may be conditional on our current understanding if needed.
Overall, the algorithm in Theorem 3.3 is interesting and clever, but it leaves the impression that something even cleverer can be done. It does not use the special properties of the functionals that give finite type invariants. It also seems not to exhaust common techniques from similar counting problems. Indeed, the authors announce: "in a later publication [BNBNHS] we plan to improve the bounds in both theorems by substantial amounts." Such an improved paper may be suitable for JACT. At the same time, it raises the question why publish these weaker results by the same authors separately. Therefore, I recommend rejecting this paper, unless these concerns are addressed.

REPORT ON THE PAPER

YARN BALL KNOTS AND FASTER COMPUTATIONS

by D. Bar-Natan, I. Bar-Natan, I. Halacheva, and N. Scherich

In this paper the authors question the computational complexity for computing a knot invariant, say ζ, from a (2D) knot diagram as supposed to a 3D representation of a knot.

More precisely, the authors consider knots as knotted tubes densely packed to resemble yarn balls, as supposed to the usual diagrammatic representations resembling pancakes. Then a yarn ball of diameter L has a volume V analogous to L^3, and the 3D computational complexity of ζ is computed using V, while the 2D computational complexity of ζ is usually computed on tne number of crossings n of a diagram.

Then they argue that for some invariants ζ (such as the linking number and in general finite type invariants) the computational complexity for computing ζ, even employing the best (currently known) 2D techniques, the 3D computational complexity $C_\zeta 3D$ would be always « than the 2D computational complexity $C_\zeta 2D$. Namely, for ζ a finite type invariant they obtain the results in Theorems 3.2 and 3.3, using the well-known Gauss diagrams. For the proofs they first describe a process of converting an oriented yarn ball knot of length/volume V to a grid knot (and vice versa) and this process costs a negligible amount of computation time.

The paper goes on with discussing the 3D computational complexity versus the 2D for stringy quantities such the hyperbolic volume and the knot genus, for which the 3D option should be expected to give better results.

The paper is original and very interesting for the mathematical community of knot theory. Based on the above, I am happy to **recommend it for publication in APCT**.

Below are some comments to be considered by the authors.

Comments / suggestions:

p.2, l.23 & l.31: the length 1 in the uniform width 1 of the knotted tube maybe confused with the length 1 in 1x1 squares.

p.2, l.38: the use of the symbol '»' really depends on assuming V sufficiently large. No?

p.5, l.34: For the blue lines might be better to use a vertical symbol instead of the horizontal one.

p.5, ll.45,46: better phrase the linking number with parentheses: is one half (the number of positive crossings minus the number of negative crossings).

p.5, l.48: In arguing that the linking number is 2D computationally more costly, the the total number of crossings, n, is used and not the number of the mixed crossings only, say m. So, it needs to be argued that the computational time is really analogous and not much less. Unless you include also in the computational time checking whether a crossing is mixed or not.

p.14: In several references capital letters of names are small.

Reviewer #1's comments (on revised version):

In my previous report I mentioned that the main definitions in pages 1-5 are not rigorous, and mentioned "C3D" as one example. In the revision, the authors minorly address this issue by removing the term C3D from one of the three places where the main result Theorem 2.1 is stated. They also move the "conversation starter 1" defining C3D one page forward, framing it as "discussion", without any further changes.

I have read the paper again, and I still believe that a higher level of mathematical rigor is crucial. It may be helpful if I mention a few more detailed examples of non-rigorous writing, that demonstrate the kind of revision needed throughout the introduction. This list is not exhaustive.

1. The definition of the so called "is the same as" relation: "We write A ? B for two expressions A, B depending on n if O(A) = O(B), with the assumption that log(n) ? 1." - This is not a precise definition, with an incorrect usage of the standard big-O notation.

2. "We define a yarn ball knot to be a knotted tube of uniform width 1 that is tightly packed in a ball, as in Figure 1 (B)." Is "tightly packed" part of the "definition"? What does it mean exactly? The representation of knots as a "yarn ball" appears later in the definition of C(3D,-). The term yarn ball must have a precise (and finite) meaning because it serves as input to an algorithm in Theorem 2.1.

3. Is "V ~ L^3" (volume is diameter cubed) part of the definition of a yarn ball, or a consequence, or a special case worth discussing? My interpretation of Theorems 2.1 and 3.3 is that formally V denotes the length of a given embedded knot, without any further yarn ball assumptions. I guess so because complexity is usually measured in terms of the length of the input. However, in the free text V is regarded as volume and it is not clear if/where/why it is also assumed to be the diameter^3.?

4. What is the definition of the symbol "<<" appearing in C3D and elsewhere? Perhaps it means "=o( )", but in the comments of the other reviewer it seems to be understood differently. The author added "For sufficiently large V, V^(4/3) >> V". I cannot think of a sense of ">>" where the addition "sufficiently large" is not superfluous. A precise definition would be a better way to clarify.

5. As in the previous report, it is still not clear why the status of C3D of an invariant "depends on our current knowledge". According to my best interpretation of "conversation starter 1", the definition of C3D seems to depend on comparing the asymptotic time complexities of two well-defined function problems. Perhaps I do not think about C(3D,-) and C(2D,-) the same way the authors meant. In any case, this is a central point of the paper, and a clarification is still much needed.

Finally, unlike the first version, the authors now write that they prefer not to publish their improvement of Theorems 3.2 and 3.3, and have removed the "in preparation" reference. This somehow addresses the concern that this paper might be just a weaker version of their next one. They give the reason that the argument is "highly technical" and "would suit a different audience." In my opinion, it is ok to publish highly technical computational methods in APCT.