

Response to Reviewers

Reviewer #1

Thank you for your comments and suggestions! We have addressed each of your comments below.

Page 2

Description of data collection

I feel that one can give a more gentle introduction to an underlying problem of packing simulation here.

Mentioned that we use geometry-based particle packing simulations to generate a set of spheres within a volume and analyze the packing fraction (occupied space vs. total space).

One can also explain here or below why exactly simulations fail. Is it an underlying property of the algorithm/simulation or of this particular realization that you used? For example, <https://github.com/VasiliBaranov/packing-generation> uses for the LS algorithm the Verlet lists/cell lists optimization to find relevant particle neighbors quickly, but for very dilute packings it can miss some relevant neighbors, so the algorithm can fail due to missing some relevant neighbors during approximate search. In this case, it is the consequence of implementation.

Mentioned that the algorithm can fail due to missing neighboring particles during an approximate search.

Page 3

The failure probability dataset (`sobol_probability_filter.csv`) contains unique input parameter sets (nine variables) and the estimated probabilities that the simulation will fail at each of the two steps (force-biased algorithm and Lubachevsky-Stillinger).

It would be nice to put links to the force-biased algorithm and Lubachevsky-Stillinger papers here.

References have been added here.

Page 4

Figure 1 contains a histogram for the number of successful repeats for each parameter combination.

It took me some time to decipher this sentence and figure.

This has been updated for clarity, including an example.

Figure 2 contains the probability of a simulation failing for each of the two algorithms.

I am not sure that I understand it. If this is the probability of failure for two algorithms, there shall be two numbers..

This has been changed to: "For a given parameter set, the probability of a simulation failing is the number of failed simulations divided by the number of simulations that were run. Figure 2 contains the probabilities of a parameter set failing for each of the two algorithms (force-biased and Lubachevsky–Stillinger)."

Page 6

Figure 2. Histogram of normalized simulation counts vs. the probability of a simulation failing for a given parameter set.

Ah, so for each parameter set you track how many simulations you did and what was the corresponding probability of failure? Maybe you can mention this or explain better somehow.

That's correct. This has been updated per the comment above.

Page 7

Figure 3. Histogram of number of simulations vs. packing fraction for the force-biased algorithm or fba (blue) and Lubachevsky–Stillinger or ls algorithm (red). On average, the ls algorithm tends to have higher packing fractions with a more Gaussian-like distribution than fba.

It is a bit suspicious that the LS algorithm produces packings only above density 0.64, i.e. jammed packings only. Maybe you used the version of the algorithm that is called ls_{gd} in <https://github.com/VasiliBaranov/packing-generation> It is then slightly different from the pure LS algorithm (the compression rate decreases) and it can be explained a little in the text.

We have updated this to reflect that the algorithms were run in a two-step process: first an attempt to run FBA followed by an attempt to run LS. Even if FBA failed, LS was still attempted. We included a link to the relevant code.

In general, one could explain the underlying problem here slightly more and may structure the section "Experimental design, materials and methods" to make it easier to follow, I feel.

This has been addressed and the structure updated.

Also, it would make sense to mention <https://github.com/VasiliBaranov/packing-generation> in section Experimental design, materials and methods.

This has been added.

It would also be fair to cite the papers where the code above was introduced and used, e.g. <https://pubs.rsc.org/en/content/articlehtml/2014/sm/c3sm52959b>

This has been referenced.

We acknowledge Vasili Baranov and Robin De Schepper for help with the packing-generation codebase

It is probably more appropriate to use in acknowledgements the author name from the paper, <https://pubs.rsc.org/en/content/articlehtml/2014/sm/c3sm52959b>, Vasili Baranau.

This has been updated.