EGUsphere, author comment AC1
https://doi.org/10.5194/egusphere-2022-1243-AC1, 2022

Reply on CC1

Emily de Jong et al.

Author comment on "Breakups are complicated: an efficient representation of collisional breakup in the superdroplet method" by Emily de Jong et al., EGUsphere, https://doi.org/10.5194/egusphere-2022-1243-AC1, 2022

Dear Axel,

We welcome your comments—thank you for initiating what I hope will be a fruitful discussion! I am glad to hear your feedback, considering your experience studying collisional breakup. Allow me to respond to each of your comments individually, and then I hope we can pursue the most interesting topics further.

- Filament breakup: You are correct in observing that the breakup method we propose cannot describe the outcome of a breakup event resulting in A′, B′, and C′ droplets. Using your notation, suppose we have the breakup event: A + B ☐ A′ + B′ + C′. In fact, we only represent either A′, B′, or C′, as one of the outgoing superdroplets contains excess uncollided droplets A, leaving only one superdroplet to represent the collision fragments. If we consider a single collision, this representation is indeed unphysical. However, averaged over many collisions, sampling from the fragment size distribution (including A′, B′, and C′) many times will recover the statistics of the overall size distribution. It is not a perfect or exact solution, but it does allow us to have a probabilistic representation of breakup without creating new superdroplets. The stochasticity that goes with sampling from the fragment sizes could also be a benefit in that two simulations might produce slightly different results, giving us some uncertainty bounds on the dynamics.
-

  Thank you for drawing attention to your own implementation of breakup—I failed to reference it in the work, but will correct that in the revisions. I see that the merging of superdroplets is discussed in a related work (https://gmd.copernicus.org/articles/7/695/2014/), and that it involves creating an ordered list. I do not see any discussion of parallelization in the same work, which leads to my concern that such an operation would break the linearity that the original SDM implementation has. Is this how your computations proceed? How does it scale in practice?

  The motivation for this implementation is to maintain the key virtues of the original SDM coalescence algorithm, namely that: (i) it has linear complexity in number of superdroplets; and (ii) it is embarrassingly parallelizable (constant state-vector size,

with no data dependencies across particle pairs). It then follows that we have a crucial requirement to keep the number of superdroplets constant at all times, and may be unable to perform the merging your method requires.

- I am very grateful to you for bringing up these caveats for the Straub/Schlottke parameterizations. If anything, your note suggests that we should implement all three parameterizations and observe whether there is any difference in the outcome in the simple box test cases. That will give some indication of how much sensitivity in the parameterization our simple "uniform breakup" algorithm can actually capture.

- Perhaps the way we have presented the sampling in Appendix A is overly complicated or unnecessary, because in fact it accomplishes exactly the steps that you mention. We use the random number to decide first which mode of breakup occurs, rescale the random number, and then sample directly from the fragment size distribution. The only caveat to this is that the second step, sampling directly, requires an analytic cumulative distribution function. This is easily accomplished by some common computing packages, such as CUDA and Numba which both support the erfc function, and can be approximated mathematically where an analytic CDF is not available.

- Thank you for pointing this out—it sounds like a more appropriate limiter then would be to limit A' to (A+B) in size.

- It would indeed be ideal to have a reference model with the exact solution, perhaps following the implementation you suggested above in (2) or even using an existing superdroplet-increasing code such as yours. The one challenge that I can foresee with doing this type of comparison is that both approaches still involve sampling from a distribution, which might lead to divergent behaviors unless we average over enough superdroplets or simulation instances.

  Better yet, we could compare to a deterministic model instead, using the full fragment size distribution in any breakup step. One such option is exploring a set of cases with analytical solutions (as in https://iopscience.iop.org/article/10.1088/0305-4470/28/11/004). For other cases, a bin model would be the obvious choice, but given the caveats you warned about regarding mass conservation and closure assumptions, there may be other errors involved.

Thank you again for the detailed and thoughtful remarks, Axel, and for prompting what I hope will be an interesting discussion.

Best,

Emily