Summary

This paper presents BinMod1D, a Python package for simulating collisional coalescence and breakup processes in precipitation, with corresponding polarimetric radar signatures. The model can run as a box model, 1D steady-state, or full time/height column model.

Overall Assessment: The paper is well-written and presents valuable tools for the cloud microphysics and radar meteorology community. The work is solid and the implementation is useful. Future work demonstrating specific applications and case studies will help the community better understand where this modeling approach fits within the broader landscape of microphysical modeling tools and how it can generate the most scientific impact.

Thank you very much for your comments and suggestions. I have addressed your comments below in blue text.

Comments

General Comment on Figures 1-3: The text describing the methodology is quite dense, and while the figure annotations are good, the narrative could be expanded to better guide readers through interpreting these complex diagrams. Consider adding:

- Step-by-step walkthrough of what each panel shows

- More explicit connections between figure elements and the mathematical terms in equations

- A "how to read this figure" text, especially for Figure 1's complex gain/loss regions

- Clearer transitions from figure to text and back

The figures contain important conceptual information about the numerical scheme, but readers may struggle to extract insights without more narrative scaffolding.

Thank you for these suggestions. I have added more detail to the Figure 1-3 narrative and equations to better explain these figures and equations and how these methodologies work. I've also slightly modified Figure 1a to better illustrate how the source bins correspond to the gain region. Some of your suggestions would be difficult to directly implement. For example, it is difficult to provide step-by-step walkthroughs as the figures themselves represent a *single step* in the numerical integration methodology. However, I have added more text to better explain these figures. In particular, I added the following components:

1. I modified Figure 1 to include hashed source (i) bins for the target k bin to highlight that both i and j bins are being included in the k bin gain.
2. In the text describing Figure 1, I incorporated some of the mathematical details regarding how the x+y line relates to the x_k and x_k+1 k bin edges as well as how this creates the red “Gain” region.
3. I added parenthetical explanations in equation 9a-d which describe how each integration term corresponds to number/mass gain and loss transfer amounts.

Discussion

The paper presents a solid technical contribution, but would benefit from expanded discussion of:

- Where BinMod1D fits within the existing ecosystem of microphysical modeling tools

- Specific use cases where this approach has advantages over alternatives.

- Potential applications in operational radar retrieval development or research workflows

- Future work demonstrating community impact through real-world case studies

This would help readers understand not just *what* the tool does, but *when* and *why* they should use it.

Thank you for your suggestions. I’ve added additional text to the conclusions section which touches upon these points. I would like to point out that the original version does already have some of these elements. For example, in the current preprint version, I do mention the possibility of using BinMod1D for investigating the evolution of snow size distributions using initial PSDs taken from in situ aircraft or laboratory measurements (similar to the Field et al. 2006 study and Connolly et al. 2012 study, for example). In the conclusions section of the current preprint, I also provide a quite detailed BinMod1D example which describes a potential application of radar retrieval development as well. However, I have added a few additional sentences in the conclusions that address these suggestions:

“BinMod1D is a simple, economic spectral bin microphysics model that should be accessible for those familiar with the Python programming language. **Other similar Python packages like LCM1D and PySDM use Lagrangian depictions of the PSD evolution rather than the typical Eulerian description given by traditional spectral bin microphysics schemes. In this way, BinMod1D represents a much more traditional approach that is more similar to the widely-used Fast Spectral Bin Microphysics scheme (FSBM; Khain and Lynn 2009) which is available in the Fortran-based Weather Research and Forecasting (WRF) model. BinMod1D provides an expedient way to not only generate box model or 1D column model PSD evolution due to**

coalescence and/or breakup, but also to easily incorporate various habit parameters, numbers of distributions, and interaction kernels. Additionally, BinMod1D, unlike other similar Python packages, allows for direct, explicit simulation of steady-state vertical radar profiles. This article provides only a small set of examples for using the model. However, due to its flexibility, BinMod1D can be customized for a variety of different purposes. For example, users could use BinMod1D to perform in-situ-type bin model simulations similar to Field et al. (2006) where actual in-situ-derived particle size distributions are used to initialize the bin model and investigate the theoretical effects of aggregation (and/or breakup) compared to measured distributions taken from such Lagrangian aircraft spirals. Similarly, users could also develop theoretical particle size distributions valid for laboratory studies such as done in Connolly et al. (2012)."

Minor Comments

- Line 127: Missing text - should read "model, due to" (I think)

From the online preprint, I believe the line in question is this:

"Therefore, the ability of BinMod1D to directly simulate steady-state vertical profiles drastically decreases the computational time compared to running the 1D column model to its steady state as done in several rainshaft studies (e.g., Prat and Barros, 2007b; Prat et al., 2012; Kumjian and Prat, 2014)."

If I'm looking at the correct line here, I believe what I have written is correct. The point here is that BinMod1D allows users to directly calculate steady-state (in height) profiles rather than running a time/height column model out to its steady-state solution (using a sufficiently long time period). Please let me know if this is not the line in question or if there was a different concern here.
