Authors' Response to Referee Comment #1

on Manuscript "CLEO: The Fundamental Design for High Computational Performance of a New Superdroplet Model" by Clara J.A. Bayley et al. (2026)

Title	CLEO: The Fundamental Design for High Computational Performance of a New Superdroplet Model
Author(s)	Clara J.A. Bayley et al.
MS No.	egusphere-2025-4398
MS type	Model description paper

We thank Referee #1 very much for taking the time to review our manuscript and suggest improvements. We found their comments constructively critical as well as helpful for making our manuscript more accessible to a wider audience, most of all regarding our discussion of Monoids. We address the general comment in the following section and thereafter the minor comments and revisions.

Please note the updates to the dataset for this paper (https://doi.org/10.17617/3.LNRKSJ) are currently viewable from this link whilst they await publishing:
https://edmond.mpg.de/previewurl.xhtml?token=3eecde11-6c96-473f-9037-bfb58344d59b.

Response to the General Comment

The general comment made was the following: *This is an interesting manuscript, but very computer science oriented. It describes a framework for a new super droplet code. There is not much scientific content. Normally this would likely be normally this is supplementary material to an actual set of scientific simulations, even for Geoscientific Model development. It seems basically a proof of concept. It almost seems like this needs to be in a computer science journal. I am not sure how GMD should treat that. It's up to the editor on that front. There is nothing really wrong about the manuscript as a description of the technical aspects of a framework for a new superdoplet code. I also note that much of the computer science stuff is beyond my expertise. I was hoping for more scientific content. I'm not sure how ready for scientific content this code is.*

We understand this Referee is uncertain whether the content of this manuscript is relevant for GMD; nevertheless we feel strongly that it is. Indeed we intentionally wrote this paper to cover only the fundamental design (i.e. "the technical framework of our code"), and we kindly refer our readers to its companion paper: "Cleo: The Numerical Methods of a New Superdroplet Model including a Droplet Breakup Algorithm" by Bayley et al. (egusphere-2025-4399), for the scientific capabilities of our model. Our reasoning for dividing the scientific content and the computational into two separate manuscripts was exactly because the papers cover very different aspects of the model and therefore appeal to very distinct audiences. This paper appeals to the more computationally grounded audience, interested in the technical details and challenges of scientific programming for superdroplet models, whereas the companion to this paper is targeted towards the "scientist user" of the model, who is instead interested in the cloud microphysics which Cleo can model. Both papers are necessary to form a complete description of Cleo as a new superdroplet model. We feel our decision to describe Cleo's computational structure in such detail is well supported by the response of Referee # 2, which, in our opinion, demonstrates there is an audience who are greatly interested in the computer science underlying our model. We also agree with Referee #2 that the topics we cover in this manuscript are often missing in the cloud microphysics community's literature on model development, although they are of clear value and match the scope of GMD.

Response to Minor Comments

We thank the Referee for the minor questions and clarifications they posed. We have addressed all the "minor comments" individually in the manuscript, and provide additional explanation for what we changed in the following.

- *Page 1, L16: 'obscurity' is not the right word. Uncertainty?* We have deleted the word, the text now reads (P1 L17-20): "With the advent of Global Storm Resolving Models (GSRMs) which remove the need to parametrise convection, ~~obscurity in~~ cloud microphysics is now one of the leading sources of uncertainty in global models too (e.g. Miyakawa et al., 2014; Stevens et al., 2020; Suematsu et al., 2021; Bao andWindmiller, 2021; Lang et al., 2023; Takasuka et al., 2024; Naumann et al., 2025).".

Page 2, L41: However, SDM still requires use of collision / collection kernels, which are uncertain. . . .might need to note that. Bin schemes have the same issue (and bulk schemes don't represent this at all).

Yes we completely agree with this comment, and we discuss this extensively in the companion paper to this one: egusphere-2025-4399, e.g. allowing two different treatments of collisions (P5 L126-132), different kernels within these treatments (P6 L143-150), and also different outcomes (P7 L157-162). We have also amended the text in this manuscript to state (P2 L48-50): "Although this does not eliminate the epistemic uncertainty caused by insufficient knowledge of microphysical processes and condensate attributes, SDM's convergence property makes its physical interpretation both conceptually elegant and straightforward. This in turn helps us explore such knowledge gaps.".

- *Page 3, L65: Define CLEO with first use in text as well as the abstract.*

  In response to a comment by Referee #2, we've decided to remove the CLEO acronym entirely and rename "CLEO" to "Cleo". In that response we explain "[...] We think the acronym we gave Cleo was one of the key reasons the expectations [the manuscript set] did not align [with its content] and upon further consideration we also find this acronym not only unhelpful in this context but also restrictive for Cleo's future development (we may expand Cleo beyond cloud microphysics and/or use Cleo on non-exascale systems). We've therefore reverted to using Cleo as a proper noun, as it was originally intended to be, in tribute to the real-life "super-women", Cleopatra and the anonymous mathematican whose pseudonym was Cleo."

- *Page 3, L68: Define exascale computer*

  Certainly; we have added (P3 L75-76) "[...] exascale high-performance computers, computing at least 10^{18} floating-point operations per second.".

- *Page 3, L77: What other aspects of performance?*

  We now state them explicitly; we have added (P4 L94-96) "[...] whilst compromising on other aspects of aspects of performance, namely Single Instruction, Multiple Data (SIMD) parallel processing and scalability with increasing number of superdroplet attributes.".

- Here we respond to the two comments regarding monoids collectively. These were:

  - *Page 3, L82: Please describe what a monoid set is. Maybe with an example. See below.*
  - *Page 3, L82 and Page 9, L230: For the non-mathematically inclined. Can you give a simple example of a monoid? The description is not that clear. What are some binary operations? What is a semi group? What is an identity element?*

  We have re-written the first paragraph of the monoids section to address these comments. At the start of this explanation we rephrase the mathematical description of a monoid in simpler terms and thereafter we include the tangible example of acrylic paint. The paragraph now reads (P11 L291-297): "Cleo uses monoids to enable model flexibility without added run-time from conditional code branching. In abstract mathematics, a monoid is a closed set with an associative binary operation and an identity element. Put simply, monoid sets are composed of elements (members) which result in another element of the set when they're added pairwise together, and the identity element is defined such that the outcome of its addition with another element is just that other element unchanged. Acrylic paint is one example of a monoid set, whereby different colours are the elements of the set and transparent acrylic paint is the identity element. Mixing two acrylic paints together is their associative binary operation (in this special case, also commutative).". We have removed the word "semigroup" as we realised it is more confusing than helpful to non-expert mathematicians / set theorists.

- Here we respond to the two comments regarding advection collectively. These were:

  - *Page 7, L191: How will the array of super droplets with different grid positions be handled by advection? Is this a problem for efficiently if that part of the code wants to loop over grid boxes?*
  - *Page 7, L191 and Page 9 L227: Is the basic intent then to run the SDM on a different grid than the dynamics/? See comments above about advection. Can the SDM do its own advection if 1 way coupled?*

  We have added text clarifying how Cleo handles (superdroplet) advection in P8 L217-220 "Whereas Cleo advects superdroplets according to the fluid flow (wind fields) itself, to advect thermodynamic fields (temperature, pressure etc.) Cleo must be coupled to a host dynamical driver capable of advection. However the domain decomposition and computational resources for the dynamics can be independent from those used by SDM." as well as in P10 L275-279: "Cleo can enact microphysical processes on the superdroplets

in each grid-box and it can advect superdroplets according to Figure 3: first updating their coordinates and SD-GBX-Indexes and then re-ordering the superdroplet arrays. The numerical methods for microphysics and for updating superdroplet coordinates are provided in Bayley et al. (2025). Whilst Cleo can thus advect superdroplets, to advect thermodynamic fields as per fluid dynamics, Cleo must be coupled to a dynamics-solver with a fluid-dynamical core."

Specifically regarding the question *"Is this a problem for efficiently if that part of the code wants to loop over grid boxes?"*, we kindly refer the reader to the changes we made in response to the comment by Referee #2 regarding Kokkos Thread Parallelism.

Specifically regarding the question *Is the basic intent then to run the SDM on a different grid than the dynamics/?* Yes one part of the intent is that SDM can run on a different grid than the dynamics. However the other part, perhaps more important too, is that Cleo can run concurrently to the dynamics and hence Cleo can have a different domain decomposition. As stated in P10 L271-272: "Cleo is designed to run concurrently to a host dynamical driver called a "dynamics-solver" and exchange information with it via a "dynamics-coupler"" and hence (P11 L286-289) "In general, using MPI means Cleo and the dynamics-solver do not have shared memory and that their domain decompositions are independent. Whilst this can result in costlier communication, it also maximises our freedom to optimise the load balancing and so economise the allocation of computer resources.". The option for using a different grid as well as domain decomposition is in some-ways a bonus, which may allow the numerical methods for dynamics and SDM to be more accurate or easier to compute, as stated in P11 L284-286 "Using different grids is advantageous because it enables the dynamics-solver to compose the domain in the optimal way for its fluid-dynamics. Meanwhile Cleo can compose the domain more favourably for SDM, for example using a nested grid, or grid boundaries which reduce grid-box volumes and/or simplify the numerics of superdroplet motion.". (The drawback being that you need to interpolate between the grids during communication, but fortunately the YAC coupler does that well for us.)

- *Page 11, L289: But what if process A is estimated with a rate that would result in say complete removal of drops, and then halfway through that A step process B depletes more? How do you harmonize different process rates with different timeteps?*

  In P12 L312-317 we have added text to make it clearer that the monoids are associative but not necessarily commutative: "Note that since monoids are not necessarily commutative, "AB" is allowed to cause different microphysical outcomes to those of "BA" — in this example, the growth of superdroplets by condensation before collisions can result in different outcomes to collisions followed by condensation (i.e $AB \neq BA$). [...] [Cleo creates] one final object, Z, whose "do_microphysics" function is an assembly of all the microphysics we desire (again, associatively but not necessarily commutativity).". In your example, supposing at the current time-step both A and B are "on-step" i.e. should be called, then if process A is enacted first there will be the complete removal of drops such that when B is called it cannot remove any more. If process B where however enacted first, some droplets will be removed by process B and then A will remove the rest afterwards. To get the desired outcome of microphysics, be that A then B or B then A, one must choose to combine A and B in the correct order. This is irrespective of whether or not they have different time-steps, except that perhaps users may desire that processes with shorter time-steps are computed first (in your example, B then A not A then B).

- *Page 15, L319: How does Figure 5a relate to figure 5b? Not clear how figure 5b plugs into 5a/*

  We've added dashed arrows to connect Figure 5a and 5b together so the part where Figure 5b plugs into Figure 5a is now more easily identifiable.

- *Page 16, L360: This drop concentration is quite high and represents very polluted conditions. 100cm-3 would be more reasonable over land. Does that affect the results? You likely would get more precipitation. Are you trying to delay it?*

  We chose this aerosol number concentration because it is a reasonable value for marine aerosol conditions (Lohmann et al., 2016), although as you rightly point out it delays precipitation and would represent very polluted conditions over land. Delaying precipitation helps to keep the superdroplets in the domain longer and so have more time-steps of the simulation to average over when measuring the performance. The performance results, once normalised by simulated time-step as we have done, would be negligibly affected by lowering the droplet number concentration, we would just have less time-steps to average over.

- *Page 30, Figure 6: what time in the simulation is this? 80 min? Also, why does # superdroplets = # grid boxes? One per grid box?*

  Yes this is the cumulated trajectories of the random sample of superdroplets after 80 mins. We've clarified this in the figure caption which now reads: "An example of the performance tests with 16384 grid-boxes. (a-d) The initial conditions, (e) The growth and trajectories of a random sample of 500 superdroplets between

1500 m$< x <$ 3000 m up-to 80 mins for the test case with 4.194×10^6 superdroplets overall." as well as in the main text (P19 L459): "An example of the superdroplets' evolution up-to the end of the simulation is shown in Figure 6e.". Thank you for spotting the error in the figure caption regarding the number of superdroplets, we have corrected this and apologise for any confusion it may have caused.

Additional Minor Revisions

As well as taking on the feedback of the Referee, we have made some additional minor changes to the original manuscript listed here:

- P1, L19-20 and P2 L31-32: corrected references.
- P3 L89 and P20 L518: clarified meaning of the dependence of Cleo's MPI performance on the host-dynamical model by changing "the host dynamical model" to "the fluid-flow given by the host dynamical model".
- P12 L335-336: added an additional point regarding the benefits of templated code: "This allows compilers to efficiently inline and optimise the used code, whilst at the same time increasing code locality by not compiling unused code".
- P17 L410: "4 byte" rephrased "4-byte types".
- P21 L557: "minimum bound" rephrased "conservative estimate".
- Code listings: formatting adjusted to allow line-breaks.
- Code and data availability: citation formatting corrected.
- Acknowledgements: added another funding source and referees.
- References: Updated the companion manuscript citation (for egusphere-2025-4399).

References

Lohmann, U., Lüönd, F., and Mahrt, F.: An Introduction to Clouds: From the Microscale to Climate, Cambridge University Press, ISBN 9781139087513, https://doi.org/10.1017/CBO9781139087513, 2016.