This document contains the Reply to Reviewer 1, the Reply to Reviewer 2 and the revised manuscript with marked-up changes.

We want to thank reviewer 1 for his/her careful reading of the manuscript, the constructive and valuable comments. This document contains point-to-point replies to each point the reviewer made. The reviewer's comments are written in normal, our replies in bold italic font. Line numbers preceded by an exclamation mark refer to the original manuscript, those without to the revised one.

The authors compare three different Langrangian Cloud Model (LCM) implementations with a focus on collection using three different collection kernels. Analytical solutions as well as previous bin model results are used as references. Additionally, sensitivity of the LCM implementations with respect to the initialization of the simulation particles (SIPs) as well as to different numerical features (resolution, time step, ...) is tested.

This results in a large amount of model runs with a great variety of possible parameter and configuration combinations which sometimes makes the manuscript difficult to read.

General comments

Each of the LCM implementations shows rather strong shortcomings:

• RMA cannot deal with realistic kernels (Long, Hall) and shows spurious oscillations.

• AIM systematically underestimates the collisional growth.

• AON always needs an ensemble of at least 50 realizations to reach a representative average result for the final drop size distribution since individual realizations deviate considerably from the average (in contrast to RMA and AIM). This severely limits the potential to be used in 2- and 3-dimensional models with a large number of grid points.

We agree with the reviewer's opinion on RMA and AIM. For AON we are not as pessimistic as the reviewer regarding its use in 2-D and 3-D models. It will not be necessary to run 50 realisations in each grid box. In such cases, the averaging will occur over grid boxes with similar atmospheric conditions. We made a similar experience at least in simulations of contrail-cirrus, where we tested the NSIP-sensitivity of the deposition/sublimation process (see section 3.1 in Unterstrasser & Sölch, 2014). We found that very few SIPs per grid sufficed to reach convergence even though the few SIPs in a single grid box could not realistically represent a smooth size distribution. Smooth size distributions can be derived only for larger volumes of air when more SIPs are taken into account. This explanation is now given in section 4.1.

It will be interesting to see how AON performs in 2D/3D-setups. This will be the next step in an upcoming study.

Additionally, sensitivities with respect to initialization of SIPs are shown to be high at least for some configurations. This problem is discussed towards the end of the manuscript where also more mature drop size distributions are used for initializations.

Within a full microphysics description including drop nucleation and condensational growth, it should be harder to control the DSD at the moment when collisions become important. This discussion

We agree and it will be interesting to see how the algorithms behave in such a setup and what extensions may be necessary to guarantee optimal SIP weights. SIP splitting and merging as discussed in Unterstrasser & Sölch, 2014 may be an option.

should be extended. Compared to spectral bin models the accuracy of all LCM implementations shown seems to be lower with at least comparable computational costs. One could conclude that LCMs are of no practical use. Nevertheless, LCMs are valuable tools. Please discuss critically advantages and disadvantages of LCMs.

In the Introduction we add a paragraph mentioning the most important advantages/disadvantages of LCMs: "Due to their specific construction, LCMs offer a variety of advantages in comparison to spectral-bin and bulk cloud models. Their representation of aerosol activation and subsequent diffusional growth follows closely fundamental equations and avoids therefore the possible perils of parameterizations (e.g. Andrejczuk et al. 2008, Hoffmann 2016). The same applies for the representation of collection or aggregation, which is based on the interaction of individual SIPs. Accordingly, LCMs approximate pure stochastic growth (e.g. Gillespie 1975), which is the correct description of collection/aggregation within a limited system of interacting particles and results in the SCE, which is used as the basis for spectral-bin and bulk models, if the system becomes infinite (e.g. Bayewitz et al. 1974). Moreover, LCMs do not apply the finite-differences method to compute microphysics. Accordingly, LCMs are not prone to numerical diffusion and dispersion, and do not suffer from the numerical broadening of a droplet spectrum, which can affect spectral-bin cloud models (Khain et al. 2000). Finally, LCMs enable new ways of analysis by the tracking of individual SIPs. They can be used to reveal the origins of droplets, as well as conditions associated with their growth (e.g. Hoffmann et al. 2015, Naumann and Seifert 2016). The largest disadvantage of LCMs, so far, might be their relative novelty due to their higher computational demand. Many aspects of this approach have not been validated adequately or can be improved. For the process of collection/aggregation, this study will offer a first rigorous evaluation of the available numerical approaches. "

It is clear, that the present study is a first step of evaluation. Our next step is the AON evaluation in higher-dimensional tests. For this reason, we do not want to speculate too much about the performance in such applications. The initialisation of SIPs differs from model to model and/or application to application. Constant weights approaches, as used in several studies recently, greatly deteriorate the collection treatment. Those studies initialise the SIPs in the beginning of the simulation as CCN. Contrary to this, the LCM by Sölch and Kärcher, 2011, initialises SIPs only during cloud formation with varying SIP weights.

Regarding the computational costs no final conclusion can be made at the present stage. At least for localised cloud objects like contrails as studied with EULAG-LCM, the computational costs of LCMs and bin models are comparable. A bin model would carry out the same amount of computations in any grid box, whereas in LCM approaches no computations are carried out in the ice-free grid boxes (which can be the majority). On the other hand, this might not necessarily result in a model speed-up due to load-imbalancing.

The quality of some figures is poor. Most of them are too small, lines are too thin and sometimes too many. Specific comments are given below.

All figures have been revised.

Specific comments

l. 154-157: What is the reason to switch from mass doubling (which is often used) to a tenfold increase as a basis for bin resolution?

There is no particular reason besides personal flavour. A simple conversion formula is given in l. !157.

l. 238: If the probabilistic version of the singleSIP-init is used, dots are not distributed uniformly!

Strictly speaking, the dots are uniformly distributed only for the deterministic version. We hope that saying "homogeneously distributed" is an acceptable expression.

Fig. 1: upper left and below: difference between red and green lines is misleading since the higher density of dots wrt the x-axis is not resolved.

In the updated figure, κ is reduced such that the dots appear less connected.

upper right: threshold radius line barely can be seen; lines for alpha-values are also misleading, can be confused with legends. Values should read N 10α .

Thank you for spotting the inconsistency with the usage of α. We improved the figure.

Last but one row: Is there a systematic difference between the symbols and the lines due to plotting issues? If not, a better initial agreement should be reached. Cp. also l. 268-270.

The reason for the systematic difference between the symbols and the lines is the following: The data of the reference solution use a finer bin grid. Now we use the same κ as for the plots with the symbols and the agreement is better. As noted in the manuscript, we use $\kappa_{plot}=4$.

Algorithm 1: What do k++ and i++ stand for? loops over k/i?

This is C++ style for "k = k +1". We replace those expressions.

Algorithm 2: l. 13: Please exchange gain and loss term due to consistency with l. 12 and eq. (22)

Thanks for this hint. Corrected.

Fig. 3: Top: It is difficult to see what happens in the left part of the spectrum (<20 μm). Bottom: It is confusing to normalize one ratio to t=0 and the other one to t=3600. Please redo the black curve with vi (t = 3600)/vi(t = 0)

The motivation behind the definitions was that both fractions are >= 1 and both curves fit better in the same plot. We changed the description of the plot around line 473 to eliminate this pitfalll.

l. 559 and Figs. 6, 9, 11, 13, 15, 17, 19, 22: The third moment λ3 should not be shown in the figures since the behaviour is very similar to λ2 (which should be stated in the text). The space saved can be used to extend some other figures in order to improve their readability.

We remove all panels that show the third moments.

Fig. 5: Use full lines with enhanced line thickness for RMA results and dotted (or dashed) lines for analytical solutions. Otherwise, all plots look identical at first glance.

We changed it in all plots showing DSDs. Now the solid lines show our results, the dotted lines show the reference solution.

l. 564-575: The discussion of the RMA Golovin results is very short and misses several aspects, e.g.: Why are the results for RedLim worse than the regular ones? What are the reasons for the relatively large differences between the two OTF versions?

We expanded the description of the results in section 3.1.

Fig. 6: Are there any lines missing? Variation of η only for κ = 60 in the left and the middle column? No κ = 60 for OTF at all? Which lines fall together and which runs are carried out at all?

We added the necessary information in the caption to avoid any confusion. The variation of η was indeed only shown for κ =60. And in the right panel only four simulations were shown. We redesigned the plot and the selection of simulation at display changed slightly.

Fig. 7: see Fig. 5: full lines for the RMA results.

Done.

l. 595: Compared to the regular version (and to bin model results) I would not call the RMA RedLim results "perfect". The same holds for the OTFs results; only OTFl is almost "perfect".

We agree and reformulated the paragraph in section 3.1.

Figs. 8, 10, and 12: see Fig. 5: full lines for the AIM results.

Done.

Fig. 14: see Fig. 5: full lines for the AON results. Results plot for disregarding self-collections is missing.

Good point. We decided at some point to leave out the third panel and forgot to change the caption.

l. 739: Is this restricted to AON results or do the other methods show similar robustness wrt to the small tail of the distribution? In reality very small drops similarily do not contribute substantially to the growth of the large mode due to their low number and small individual mass. This should be reflected in model sensitivities.

This insensitivity to the small tail is even more pronounced in the AIM algorithm. This was already mentioned in the original manuscript around line !642.

l. 746: This is in contrast at least to the Golovin RMA results. Why is it reasonable for AON? Is this due to the lower number of collision events realised because of the probability restrictions?

Yes, this is the reason. We added an explanation around line 815.

l. 766: When pcrit is smaller, less collection events can be expected (see lines 469/470). A spread in v-values leads to smaller and larger v-values. Does this mean that the largest v-values are responsible for the enhanced collection?

The sentence contains a typo. "p_{crit} is smaller" must be replaced by "p_{crit} is larger". A combination of a small v SIP and a large v SIP leads to a large p_{crit}.

l. 902ff: It should be critically mentioned that AON always needs an ensemble of at least 50 realizations to reach a representative average result for the final drop size distribution since individual realizations deviate considerably from the average (in contrast to RMA and AIM). This leads to a large effort in terms of computational resources.

See our response in the beginning of this reply. We included our thoughts in section 4.1.

Technical corrections

Thank you for spotting all those typos and inconsistencies. They have been corrected.

l. 18: ... are important processes ...

Table 1: mean mass: M/N

Fig. 1 caption: alpha should be (-2, -3, -7)

l. 256: values of N 10α ...

l. 276: However, it is ...

l. 371: rather "proposed" than "discussed"

l. 410ff.: The terms "larger SIP" and "smaller SIP" are used here with the meaning "SIP with larger/smaller drops(=higher average drop mass)". Please define whether "large SIP" indicates large drops or a large number of drops within the SIP (cp. l. 510).

We intended to use large/small SIP throughout the text in the sense you mention. We added a sentence on the terminology around line 120.

Fig. 3 caption: ... function of their initial radius ... Please add that it is an AIM simulation.

l. 434: ... of each droplet within the SIP ...

l. 435: Figure 3

l. 510: In contrast to l. 410ff. smallest SIP refers to the size of the droplets not the weighting factors;

The description is correct the way it is. In both paragraphs small SIP refers to a SIP with small droplets and the terminology is consistently applied.

Fig. 11 caption: ... black curves with triangles ... green lines for vrandom; α should be (-2, -3, -7)

l. 658: green lines

l. 792: check the meaning of "large SIP", also l. 824 "heavy SIP"

Again large SIP refers to a large single droplet mass. Heavy SIP are SIPs with large total mass $\chi = \nu \mu$.

We want to thank Anna Jaruga for her careful reading of the manuscript and her constructive and valuable comments. This document contains point-to-point replies to each point the reviewer made. The reviewer's comments are written in normal, our replies in bold italic font. Line numbers preceded by an exclamation mark refer to the original manuscript, those without to the revised one.

The Authors evaluate three algorithms for representing collisions in Lagrangian cloud microphysics schemes that are available in the literature. The design and some implementation details are discussed. The accuracy of the three collision algorithms is compared against the analytical solution of Golovin kernel and bin solutions of Long and Hall kernels. A very wide parameter space is investigated. The manuscript also tests three different initialization techniques for the Lagrangian cloud microphysics schemes.

The work presented here is very useful. Lagrangian schemes offer very detailed representation of microphysical processes in clouds. Yet, because the Lagrangian methods are new, they have not been sufficiently tested in cases relevant for numerical simulations of clouds. The topic of the presented work is therefore very interesting and fits well into the scope of the GMD journal.

General comments

The presentation of the design of the three different collision algorithms is done very well. Figure 2, combined with the detailed description of collision algorithms and the "hypothetical algorithm", clearly shows the differences in treatment of collisions inherent to these three algorithms. Also, the presentation of the three different initialization procedures is done well. The Authors did an immense job at testing different algorithm options and simulation parameters. The Authors have tested 3 different algorithms ("remapping" (RMA), "average impact" (AIM) and "all-or-nothing" (AON) algorithms), used 3 different test cases (Golovin, Long and Hall kernels), 3 different initialization procedures and many different collision algorithm options and simulation parameters. The Authors have tested a big parameter space and it is a big achievement of the presented work. However, the presentation of the results from this set of tests could be improved. In my opinion the big number of figures showing results from many combinations of simulation options makes the manuscript difficult to read and pinpoint the interesting and beneficial parameter combinations. Instead of providing a report from many test simulations that were made, a more concise summary of obtained results would be more beneficial and easier to comprehend for the reader, in my opinion. In general, the quality of many figures in the manuscript is poor and sometimes makes them. The final analysis of the accuracy of the three tested collision algorithms is very critical, witch is a good aspect of the manuscript. The Lagrangian schemes are free of many numerical limitations of the bin schemes, but they do introduce new numerical challenges that need to be addressed. The big number of tests performed by the Authors allows a detailed analysis of accuracy. The final discussion of the collision algorithms could also underline some advantages of the Lagrangian schemes.

Overall, the manuscript discusses an interesting topic and provides a wide variety of tests. The corrections suggested in the following part of my review are minor and focus mostly on improving the figures.

Specific comments

1. "Too many" figures

As stated before I think that the Authors did a tremendous job implementing the three algorithms and then testing them in such a large variety of simulations. Nevertheless, in my opinion, some parts of the presentation could be improved by removing figures and providing instead a summary of the obtained results.

We agree that there are many figures.

Below I'm including some suggestions on how it might be done. For the sake of completeness and some potential future comparisons with other algorithms I would suggest moving some figures to electronic supplement. Such supplement could contain all figures, data needed to plot them and (if the Authors are willing) the scripts used for plotting. This would enable other

We created a supplement that contains a systematic and comprehensive collection of figures (showing around 100 sensitivity tests). The standardized figures show the temporal evolution of N_{SIP} and the moments 0, 2 and 3 (analogous to the plots shown in the original manuscript version). This collection discloses in detail the behaviour of the three algorithms, will facilitate future comparisons by fellow LCM developers and guide them. The underlying data (150 RMA, 270 AIM and 400 AON simulations) are not included in the supplement. For many simulations this makes no sense, as the algorithms obviously do not produce optimal results or the results are similar to basically identical. Our data can be obtained by request. Regarding the well-established Bott and/or Wang solutions, both scientists are happy sharing their data and we recommend addressing them directly.

Lagrangian scheme users to test their own implementations and then easily plot their own results against the tests performed by the Authors. A good example of such electronic supplement is in the Lauritzen 2014 GMD paper (doi: 10.5194/gmd-7-105-2014). Note, that such supplement does not demand publishing the actual code of the three algorithms but only simulation results. This is easier to do and to document.

List of figures that could be moved to supplement:

• 6, 9, 15: Both AIM and AON algorithms do not change the number of SIPs (NSIP). Maybe stating in the legend what was the number of SIPs used is enough and the first row of plots could be redundant. For the RMA algorithm it would be more beneficial for me to provide the actual number of the additional SIPs needed (for example as a % of the initial number of SIPs). For the size distribution moments it would be more beneficial for me to introduce some measure of error and then report the error value for different combinations of parameters that are tested. – In most of the plots the lines are on top of each other and are therefore not readable. A table of error values would be easier to read.

Our results have shown that the representation of collection within a DSD depends significantly on the initial DSD, its representation by SIPs (mass and weighting factor), the number of SIPs itself, the simulated time, as well as the applied collection kernel. Due to this strong case dependency, a quantitative error measure might favour the misuse of our results if there is no comparability. (Example: I used 200 SIPs and therefore the error due to collection should be 5 %.) Therefore, we would like to remain at our rather qualitative error analysis, which, however, will serve as a basis for decisions on numerical parameters for future studies. (Example: The more SIPs the better the representation.)

• 11, 19, 22 - top row and the last or second to last row: Similar to the previous comment, the NSIP is constant and therefore a clear legend instead of the first row of plots would be enough in my opinion. The behavior of the second and the third moment is very similar and I think that one row of panels could be omitted. The behavior could be only described in text. Again, introducing some error measure and reporting its value would be more informative for me. It would help to summarize all the results and enhance the comparison between different options and algorithms.

Our intention was to give to a complete overview of our results such that other developers of LCMs can best benefit from our experience. Based on both reviews, we realise that a better pre-selection of results that are shown is necessary. Hence, we follow your advices. We removed all rows showing the third moment and the SIP number evolution in AIM and AON and increase the remaining panels. We inserted a table with the N_{SIP}-values.

And the results are summarized in the newly introduced section 4.1

• 13: The behavior for Hall kernel is similar to Long kernel (Fig. 11) Perhaps stating that in text could be sufficient?

All figures with Hall simulations are moved to the supplement.

• 17: Similar to Fig. 11 and 19, maybe just two size distribution moments are sufficient? Again, some error measure would be useful.

Done.

2. Comments on figures

All the figures presented in the manuscript are too small for me to read easily. Also the font size and the line thickness is too small.

The color-coding and plot styles of some of the figures make them difficult to readme. Below I'm including a list of such figures with some ideas on how they could be improved:

• 2 – RMA algorithm: The gray font color used for text regarding contribution k is not readable. Maybe just for text a darker color could be used?

A darker color is used now and font size has been increased where possible.

• 3, 4 – top panel: The number of points and the chaotic color-coding makes it impossible for me to easily see what is happening in the left part of the plot. Reducing the number of SIPs shown, especially for the small drop sizes, would help. I would also suggest choosing line colors basing on the initial drop size rather than at random – for example http://stackoverflow.com/questions/13972287/having-line-color-vary-with-data-index-for-line-graph-in-matplotlib

I read the suggested web page. It is definitely worth to think about the colour coding. I personally favour colours, which I can assign a name ('red', 'blue'). For the suggested colour ranges, it is hard to use unambiguous colour names in the text. Figure 3 and 4 top are improved by showing fewer SIPs and we keep the original colours.

• 5, 7, 8, 10, 12, 14, 16, 18: Similar to the previous case I would suggest choosing line colors basing on the simulation time rather than at random. Especially for later figures showing oscillations for RMA or less smooth solutions for AON it would make it easier to compare different lines.

Increasing the thickness of the curves and using solid lines for the simulation results (the reference is now plotted with dotted lines) hopefully makes the plots better readable. Any of the plots you mention contains a legend with the time. So we think it is o.k. to leave the colours as they are.

• 11, 13, 19, 20, 22: Similar to the previous case, consider choosing colors basing on the number of SIPs used. It would make the first row of plots unnecessary and allow easier comparison.

The rows showing N_{SIP} have been removed from the plots.

• 16: In my opinion showing just one realisation and the average over 50 realisations could be enough. It's obvious that any realisation from AON will be burdened with irregular scatter. It's also obvious that averaging over even bigger ensemble will further smooth the solution and it could be just stated in text. Gained space could be then used to increase the size of the plots. The symbols *, + and – in the last two panels are not readable in a plot of this size and obscure the lines representing the actual size distribution.

We use a new layout. The interquartile range is now shown in a separate plot. To illustrate the probabilistic nature of AON we think it is justified to show two realisations.

• 17: The red and green colors overlay each other and make it difficult to read the figure. I'd suggest omitting one size distribution moment and using the space to significantly increase the size of the plot as well as the size of points and line thickness.

We replotted the figure and hope it is better readable now.

3. Pseudo-code listings

Please consider providing an additional caption explaining the conventions used in the listing. What lines are marked as comments and what lines are the actual pseudo-code? What does it mean if a line is written in italics, bold or in capital letters?

A paragraph on layout conventions is added at the end of section 2.2.

4. Discussion for Long kernel

The bin scheme solves the Smoluchowski equation for the number concentration function and by default should provide a smooth solution. However, the Smoluchowski equation is strictly true for infinite systems. For cases of big population of similar drops (i.e. a population of rain drops from a fully formed precipitation event) solving the Smoluchowski equation provides a good representation of the drop size distribution. In contrast, the onset of precipitation (or the "transition phase" for the Long kernel in 30-40 minutes of simulation time) might be governed by the behavior of just a few big "lucky" drops. See for instance the discussion in Lushnikov 2004 (doi: 10.1103/PhysRevLett.93.198302) and Bayewitz et. al. 1974 (doi: 10.1175/1520-0469(1974)031<1604:TEOCIA>2.0.CO;2) The bin solutions are commonly considered a true solutions during comparison studies. However it is not clear to me what volume should be used in order to ensure that solving the Smoluchowski equation is a good method for all precipitation phases. A

discussion of issues related to this topic is definitely out of the scope of this manuscript. However, could you consider adding a small warning or comment on this aspect?

We added a small comment in the introduction(around line 87) that the LCM rather solves the pure stochastic growth (Gillespie 1975) than the SCE due to the consideration of individual collisions within a finite system of particles. However, the pure stochastic growth approaches the SCE for $N \rightarrow \infty$ or, equivalently, as the number of realizations approaches infinity, which can be seen from Fig. 15.

In the summary of box model tests, could you outline in text how the difficulties encountered in the transition phase of the Long kernel actually affect the final solution at t=60 for RMA, AIM and AON for the best combination of the algorithm options?

Using the "best combination" of options, the reference solution of the spectral bin model was relatively well captured by any algorithm (e.g., by comparing the second moment for the RMA, AIM, and AON algorithms, which can be interpreted as a proxy of the largest droplets here). However, the "best combination" might not be the best combination for application in a higher dimensional model. As outlined in the Discussion and Conclusion section of our manuscript, RMA might depend on an infeasibly small time step, AIM might suffer from an inappropriate initial distribution of weighting factors. Only AON has been shown to handle most "situations" successfully. Therefore, we propose the AON algorithm for practical purposes.

Do the oscillations in RMA and scatter in AON preclude a good final solution? How accurate is the final stable and smooth solution from AIM in comparison? Is the location and value of the final maximum easily captured in AIM and AON?

These points are covered in the Conclusions of our manuscript. For RMA, the oscillations indeed preclude a good final solution if a feasible time step is used. To overcome the scatter in AON, a sufficient number of realisations is needed. And the AIM solution is stable and smooth but inherently depends on the initial distribution of SIPs.

5. Other comments

• line 221 - Could you comment on what techniques do you recommend when fighting numerical cancellation errors? What procedure was used in the current implementation?

We rephrased the paragraph. Numerical cancellation errors are smaller when expressions can be reformulated such that differences of similarly valued terms are eliminated. More on this topic can be found in classical textbooks on numerical analysis.

• line 252-253 - Could you comment on why the described behavior is considered advantageous?

We reformulated the sentence. See line 273.

• line 273 - Maybe consider stating what initialization will be used as default in the later box model tests?

Done. See line 295.

• Pseudocode for RMA, line 30 - is NSIP = ii or should it be i?

Thanks, corrected.

• Figure 3 and 4 bottom panel - normalizing once with regard to the initial condition and once with regard to the final state is confusing

Reviewer 1 had the same comment and we changed the description to use analogous definitions for both quantities (around line 473).

• line 488 - Another alternative could be to assign the product of collision to just one SIP and use the remaining SIP to split the biggest weighting factor between two SIPs. See the third to last paragraph in sec. 5.4.1 in Arabas 2015.

You refer to section 5.1.4, not 5.4.1. Yes this is also an option. It is relevant for approaches where you want to keep the total SIP number constant. Otherwise, you could simply delete one of the two SIPs and introduce a SIP splitting independently of whether it just happened that you save one SIP during an equal-v collection.

• line 535 - In my opinion performing collisions only for selected random pairs and scaling the probability is a very useful feature. It changes the asymptotic behaviour of the scheme with regard to the number of SIPs from quadratic to linear. It allows to perform simulations with a bigger number of SIPs, which increases the resolution of the obtained results. Could you consider underlying those benefits?

If some further tests are planned for the future, I would suggest adding this option to the AON implementation. On a side note, we use AON with collisions for random pairs and singleSIP init by default in our Lagrangian simulations. Out of curiosity, we ran the Long and Hall tests described in the manuscript using our default parameters. The results are similar to those presented by the Authors for AON box model tests.

It is nice to see that the manuscript was an incentive for you to carry out test runs. The computational cost of the current implementation is quadratic in terms of N_{SIP} per grid box (clearly not the total N_{SIP} of all grid boxes). Clearly, linear costs sound much better. But it is clear that this describes the asymptotic behaviour and becomes important for large N_{SIP}. My impression is that linear sampling was introduced because in the initial work of Shima tremendously many SIP were used and simulations weren't feasible at all with quadratic costs. The question will be if you need more SIPs to reach convergence when you use a linear sampling. In a full microphysical model this N_{SIP}-increase affects the costs of all the other processes as well. Nevertheless, in more complex settings the linear sampling is definitely a viable option which deserves to be tested. See paragraph in new section 4.1.

• Figures 5, 6, 7 are not averaged over 50 realisations. In contrast, the corresponding figures for AIM and AON are. Could you comment on why? Does the design of RMA algorithm guarantee no need for ensemble runs? Could the ensemble runs be obtained using one of the random initialization procedures? For Golovin kernel RMA produces good results for a single realisation, which should be underlined. If for Long and Hall kernels ensemble average does not help, it should also be underlined. Could you comment on how an ensemble average for RMA for high SIP number (for example $\kappa = 200$) for Long kernel would look? In general it was unclear for me if RMA (i) becomes unstable and does not provide a solution for Long test or (ii) is stable but generates cumbersome oscillations and wrong final solution.

Originally, we didn't actually run the RMA program for a full ensemble with 50 realisations as it was done for the other algorithms. This has two different reasons:

1. As we struggled to get realistic Long/Hall solutions with RMA, we did many tests with program versions that computed only one realisation. We never extended our implementation beyond this test framework.

2. After each time step of RMA, a new list of SIPs is created. From each bin, one SIP is generated using the exact bin values (see line 28 in Algorithm). There is no probabilistic component included in the algorithm.

To be more consistent in the presentation of the results, we programmed a version for multiple realisations and now show averages for RMA as we already did for AIM and AON. Figures with the RMA λ-evolutions, which have not been shown in the original submission, are now included in Fig.8. We changed the description of the text accordingly.

Concerning your last question: It is stable in the sense that the total mass is conserved, the higher moments do not explode and one can perform simulations over the whole 60 minutes. Indeed, Figure 7 shows size distributions for t=60min. If you call this behaviour unstable or not, depends also on how stability is defined in this context. In the end, what matters is that RMA solutions are far from the reference solution and useless (at least for feasible time steps). Before we started our evaluations, we expected that RMA is the best algorithm as we thought that the remapping is a clever approach. Hence we tested many options trying to get it "stable", but we did not succeed in this. We make this point hopefully a bit clearer by the extended Figure 7 and the added Figure 8. We also no longer use the word "stable" in the text now.

• Is it necessary to average over an ensemble for AIM?

One AIM simulation is enough. It is not necessary to average over an ensemble, as mentioned in l.!665-667. In the revised version, we repeat this finding in section 4.1.

• line 796 - For the sake of summarizing the box model simulations, could you discuss in text what was a minimum number of SIPs and a maximum timestep needed to obtain satisfactory results for the best combination of options for RMA, AIM and AON? Was the computational cost of all algorithms similar? Does it scale in the same way when increasing SIP number? Could you summarize in text how sensitive the three algorithms are to timestep?

We added a paragraph on time step sensitivity of all three algorithms and on parameter requirements for convergence in section 4.1.

• line 814 - For me, the total number of SIPs is a more intuitive parameter than kappa. Could you also state what is the total number of SIPs for κ = 20 and κ = 100?

We rephrased the paragraph (around line 935) to add the information on N_{SIP}.

Nonetheless, a small side remark: N_{SIP} is not a parameter of the singleSIP-init, it is diagnosed. This is why we stick to κ when we talk about the singleSIP-init. For ν_{draw} and ν_{const}-methods, the story is different as N_{SIP} is directly prescribed.

• line 838 - Could you comment on why the RMA is excluded in this part of the study? Are the oscillations as prominent as in the Long test scenario? Does it again fail to reproduce the bin model results at the final stage?

Good point. We carried out RMA simulation with the later initialisations. RMA fails again. The simulations are shown in the supplement.

• line 895 - Since the Authors state in line 858 that it is not clear which findings of the performed tests are most relevant for simulations of clouds, I would suggest somewhat weakening the statements about the RMA algorithm in the conclusions.

Good point.

Technical corrections

• line 142 - k! should be the factorial not faculty?

• caption of Fig. 1 - λ3 is missing

• Pseudocode for RMA, line 34 - should be "can be easily incorporated in ..."

• line 435 - should be Figure 3?

• line 560 - space missing after “per construction”.

• Figure 14 is missing the third column that should depict a version of AON without self collections.

Good point. We decided at some point to leave out the third panel and forgot to change the caption.

• line 834 - Could you rephrase the part “where the abundance of droplets larger than 10 um drops strongly”

All technical corrections have been done.

Manuscript prepared for Geosci. Model Dev.
with version 2015/09/17 7.94 Copernicus papers of the LaTeX class copernicus.cls.
Date: 2 March 2017

Collection/aggregation algorithms in Lagrangian cloud microphysical models: Rigorous evaluation in box model simulations

Simon Unterstrasser[1], Fabian Hoffmann[2], and Marion Lerch[1]

[1]Deutsches Zentrum für Luft- und Raumfahrt (DLR) – Institut für Physik der Atmosphäre, Oberpfaffenhofen, 82234 Wessling, Germany.
[2]Leibniz Universität Hannover – Institute of Meteorology and Climatology, 30419 Hannover, Germany.

Correspondence to: Simon Unterstrasser: simon.unterstrasser@dlr.de

Abstract. Recently, several Lagrangian microphysical models have been developed which use a large number of (computational) particles to represent a cloud. In particular, the collision process leading to coalescence of cloud droplets or aggregation of ice crystals is implemented differently in ~~the~~ various models. Three existing implementations are reviewed and extended, and their performance is evaluated by a comparison with well established analytical and bin model solutions. In this first step of rigorous evaluation, box model simulations with collection/aggregation being the only process considered have been performed for the three well-known kernels of Golovin, Long and Hall.

Besides numerical parameters like the time step and the number of simulation particles (SIPs) used, the details of how the initial SIP ensemble is created from a prescribed analytically defined size distribution is crucial for the performance of the algorithms. Using a constant weight technique as done in previous studies greatly underestimates the quality of the algorithms. Using better initialisation techniques considerably reduces the number of required SIPs to obtain realistic results. From the box model results recommendations for the collection/aggregation implementation in higher dimensional model setups are derived. Suitable algorithms are equally relevant to treating the ~~warm-rain~~ warm rain process and aggregation in cirrus.

1 Introduction

The collection of cloud droplets or the aggregation of ice crystals ~~is an important process~~ are important processes in liquid and ice clouds. By changing the size, number, and in the case of ice the shape of hydrometeors, collection and aggregation affect the microphysical behaviour of clouds and thereby their role in the climate system.

The warm rain process (i.e. the production of precipitation in clouds in the absence of ice) depends essentially on the collision and subsequent coalescence of cloud droplets. At its initial stage,

however, condensational growth governs the activation of aerosols and the following growth of cloud droplets, which might initiate the collection process if they become sufficiently large. Then, collection produces drizzle or raindrops, which are able to precipitate from the cloud, affecting lifetime and organisation of clouds (e.g. Albrecht, 1989; Xue et al., 2008).

In ice clouds, sedimentation, deposition growth and in particular radiative properties depend on the ice crystals' habits (Sölch and Kärcher, 2011, and references therein). Ice aggregates scatter more strongly shortwave radiation than pure ice crystals of the same mass. Recent simulation results suggest that contrail-cirrus and natural cirrus can be strongly interwoven. In the mixing area with ice crystals of both origins being present, a prominent bimodal spectrum occurs and enhances the probability of collisions (Unterstrasser et al., 2016).

The temporal change of ~~the droplet number distribution by the~~ an infinite system of droplets by collision and subsequent coalescence ~~of droplets~~ (or any other particles) is described by the stochastic collection equation (SCE), also known as kinetic collection equation, coagulation equation, Smoluchowksi or population balance equation (e.g. Wang et al., 2007). It yields:

$$\frac{\partial f_m(m,t)}{\partial t} = \frac{1}{2}\int_0^m K(m',m-m')f_m(m',t)f_m(m-m',t)\,dm' - \int_0^\infty K(m,m')f_m(m,t)f_m(m',t)\,dm', \tag{1}$$

where $f_m(m)dm$ is the number concentration within an infinitesimal interval around the mass m. The first term (gain term) accounts for the coalescence of two smaller droplets forming a new droplet with mass m, the second term (loss term) accounts for the coalescence of m-droplets with any other droplets forming a larger droplet. The collection kernel $K(m,m')$ describes the rate by which ~~a~~ an m-droplet-m'-droplet-collection occurs. Due to the symmetry of the collection kernel ($K(m,m') = K(m',m)$) the first term of the right-hand side can also be written as $\int_0^{m/2} K(m',m-m')f_m(m',t)f_m(m-m',t)\,dm'$.

For several kernel functions (mostly of polynomial form) analytic solutions exist for specific initial distributions (Golovin, 1963; Berry, 1967; Scott, 1968). The Golovin kernel (sum of masses) is given by

$$K(m,m') = b\,(m+m'). \tag{2}$$

Solutions for more realistic kernels (Long, 1974; Hall, 1980; Wang et al., 2006) and arbitrary initial distribution can be obtained with various numerical methods mainly using a bin representation of the droplet size distribution (Berry and Reinhardt, 1974; Tzivion et al., 1987; Bott, 1998; Simmel et al., 2002; Wang et al., 2007). The hydrodynamic kernel is defined as

$$K(r,r') = \pi(r+r')^2\,|w_{sed}(r) - w_{sed}(r')|\,E_c(r,r'), \tag{3}$$

based on the radius r and the sedimentation velocity w_{sed}. Parametrisations of the collection efficiency E_c are given, e.g. by Long (1974) or Hall (1980). In the above formula, the differential sedimentation is the driver of collections. No same-size collisions can occur, i.e. $K(r,r) = 0$. More sophisticated expressions for $K(r,r')$ have been derived to include turbulence enhancement of the collisional growth, which also allow same-size collisions ($K(r,r) > 0$) (e.g. Ayala et al., 2008; Grabowski and Wang, 2013; Chen et al., 2016).

Solving (1) demands simplifications in the representation of the droplet spectrum for which several numerical models have been developed. Spectral-bin models (e.g. Khain et al., 2000) represent the spectrum by dividing it into several intervals, so-called bins. This approach enables the prediction of the temporal development of the droplet number concentration in each bin by using ~~finite differences or more sophisticated numerical techniques~~ the method of finite-differences (e.g. Bott, 1998). The accuracy of these models is primarily determined by the number of used bins (usually on the order of 100), which makes them computationally challenging and prohibits their use in day-to-day applications like numerical weather prediction. Less challenging but less accurate, cloud microphysical bulk models compute the temporal change of integral quantities of the droplet spectrum ~~(e.g. Kessler, 1969a; Khairoutdinov and Kogan, 2000; Seifert and Beheng, 2001)~~ (e.g. Kessler, 1969b; Khairoutdinov and Kogan, 2
These are usually equations for the temporal evolution of bulk mass (so-called one-moment schemes), and additionally number concentration (two-moment schemes) or radar reflectivity (three-moment schemes), which describe the change of the entities of cloud droplets and rain drops (in the case of warm clouds). The separation radius between cloud droplets and rain drops depends on the details of the bulk scheme, but generally cloud droplets (up to 20 to 40 μm in radius) are assumed to have negligible sedimentation fall velocities, while larger drops, frequently subsumed as rain drops, have ~~significant sedimentation velocities~~ a sufficient sedimentation velocity to cause collision/coalescence. The interactions of cloud and rain drops are therefore described in terms of self-collection (coalescence of cloud (rain) drops resulting in cloud (rain) drops), autoconversion (coalescence of cloud droplets resulting in rain drops) and accretion (collection of cloud droplets by rain drops). A third alternative for computing cloud microphysics has been developed in the recent years: Lagrangian cloud models (LCMs). These models represent cloud microphysics on the basis of individual ~~particles~~computational particles (SIPs). Similar to spectral-bin models, LCMs enable the detailed representation of droplet spectra~~but inherently avoid spurious numerical diffusion in condensational and collisional growth usually affecting the results of~~ .

Due to their specific construction, LCMs offer a variety of advantages in comparison to spectral-bin and bulk cloud models. Their representation of aerosol activation and subsequent diffusional growth follows closely fundamental equations and avoids therefore the possible perils of parametrisations (e.g. Andrejczuk et al., 2008; Hoffmann, 2016) . The same applies for the representation of collection or aggregation, which is based on the interaction of individual SIPs. Accordingly, LCMs approximate pure stochastic growth (e.g. Gillespie, 1975) , which is the correct description of collection/aggregation

within a limited system of interacting particles and results in the SCE, which is used as the basis for spectral-bin ~~models (Andrejczuk et al., 2010; Arabas and Shima, 2013) .~~ and bulk models, if the system becomes infinite (e.g. Bayewitz et al., 1974) . Moreover, LCMs do not apply the finite-differences method to compute microphysics. Accordingly, LCMs are not prone to numerical diffusion and dispersion, and do not suffer from the numerical broadening of a droplet spectrum, which can affect spectral-bin cloud models (Khain et al., 2000) . The effect of sedimentation is incorporated in a straightforward manner in the transport equation of the SIPs and avoids numerical artefacts (Wacker and Seifert, 2001) . Finally, LCMs enable new ways of analysis by the tracking of individual SIPs. They can be used to reveal the origins of droplets, as well as conditions associated with their growth (e.g. Hoffmann et al., 2015; Naumann and Seifert, 2016) . The largest disadvantage of LCMs, so far, might be their relative novelty due to their higher computational demand. Many aspects of this approach have not been validated adequately or can be improved. For the process of collection/aggregation, this study will offer a first rigorous evaluation of the available numerical approaches.

To our knowledge, five fully coupled LCMs for warm clouds exist, which are described in ~~Andrejczuk et al. (2008); Shima et al. (2~~ Shima et al. (2009) , Riechelmann et al. (2012) , Arabas et al. (2015) and Naumann and Seifert (2015) and have been extended or applied in various problems (e.g. Andrejczuk et al., 2010; Arabas and Shima, 2013; Lee et al., 2014; Hoffmann et al., 2015). For ice clouds, three models exist (Paoli et al., 2004; Shirgaonkar and Lele, 2006; Sölch and Kärcher, 2010) which have been applied to natural cirrus (Sölch and Kärcher, 2011) and, in particular, to contrails ~~(e. g. Paoli et al., 2013; Unterstrasser, 2014; Unterstrasser and Görsch, 201~~ In the context of ice clouds and warm clouds, different names are used for processes that are similar, in particular in terms of their numerical treatment (deposition/sublimation vs. condensation/evaporation, collection vs. aggregation). Conceptually similar are particle based approaches in aerosol physics (Riemer et al., 2009; Maisels et al., 2004) which account for coagulation of aerosols (DeVille et al., 2011; Kolodko and Sabelfeld, 2003).

So far, no consistent terminology has been used in the latter publications. Various names have been used for the same things by various authors. We point out that super droplet, computational droplet and simulation particle (SIP) all have the same meaning and refer to a bunch of identical real cloud droplets (or ice crystals) represented by one Lagrangian particle. The number of real droplets represented in a SIP is denoted as weighting factor or multiplicity. Moreover, Lagrangian approaches in cloud physics have been named Lagrangian Cloud Model (LCM), super droplet method (SDM) or particle based method. In this paper, we use the terms SIP, weighting factor ν_{sim} and LCM. Here droplet refers to either real droplets or ice crystals. If we say in the following, that "SIP i is larger than SIP j", this means that the droplets represented in SIP i are larger than those in SIP j. Such a statement it is not related to the weighting factor of the SIPs.

Usually, only the liquid water or the ice of a cloud are described with a Lagrangian representation, whereas all other physical quantities (like velocity, temperature and water vapour concentration) are

described in Eulerian space (see also discussion in Hoffmann, 2016). SIPs have discrete positions $\mathbf{x}_p = (x_p, y_p, z_p)$ within a grid box. The position is regularly updated obeying the transport equation $\partial \mathbf{x}_p / \partial t = \mathbf{u}$. Microphysical processes like sedimentation and droplet growth are treated individually for each SIP. Interpolation methods can be used to evaluate the Eulerian fields at the specific SIP positions. This implicitly assumes that all ν_{sim} droplets of the SIPs are located at the same position. On the other hand, the droplets of a SIP are assumed to be well-mixed in the grid box in the LCM treatment of collection and sometimes condensation. Then, the number concentration represented by a single SIP, e. g., is given by $\nu_{sim}/\Delta V$, where ΔV is the volume of the grid box.

Lists of used symbols and abbreviation are given in Tables 1 and 2.

2 Description of the various collection/aggregation implementations

We use the terminology of Berry (1967), where $f_{\ln r}$ and $g_{\ln r}$ denote the number and mass density function with respect to the logarithm of droplet radius $\ln r$. The relations $g_{\ln r}(r) = m f_{\ln r}(r)$ and $f_{\ln r}(r) = 3 m f_m(m)$ hold. The latter designates the number density function with respect to mass and obeys the transformation property of distributions: $f_y(y)dy = f_x(x(y))dx$. For consistency with previous studies, $g_{\ln r}$ is used for plotting purposes, whereas f_m and g_m are more relevant in the following analytical derivations.

The moments of order k of the mass distribution f_m (= number density function with respect to mass) are defined as:

$$\lambda_k(t) = \int m^k f_m(m,t) dm. \tag{4}$$

The low order moments represent the number concentration ($DNC = \lambda_0$) and the mass concentration ($LWC = \lambda_1$). The analogous extensive properties $\lambda_k(t)\,\Delta V$ are the total droplet number $\mathcal{N}$, total droplet mass $\mathcal{M}$ and radar reflectivity ($\mathcal{Z} = \lambda_2\,\Delta V$). For a given SIP ensemble, the moments can be ~~simply~~ computed by

$$\lambda_{k,SIP}(t) = \left(\sum_{i=0}^{N_{SIP}} \nu_i {\mu_i}^k \right) \Big/ \Delta V\,, \tag{5}$$

where μ_i is the single droplet mass of SIP i and N_{SIP} is the number of SIPs inside a grid box. For reasons of consistency with Wang et al. (2007), we translate the SIP ensemble into a mass distribution g_m in bin representation and then compute the moments with the formula

$$\lambda_{k,BIN}(t) = \sum_{i=0}^{N_{BIN}} g_m(m_i,t) (\tilde{m}_{bb,l})^{k-1} \frac{\ln 10}{3\,\kappa} \tag{6}$$

(cf. with their equation 48).

The initialisation is successful for a given parameter set, if the moments of the SIP ensemble $\lambda_{k,SIP}$ are close to the analytical values $\lambda_{k,anal}$. For an exponential distribution (as used in this

Table 1. List of symbols.

Symbol	Value/Unit	Meaning
$f_m, \tilde{f}_m$	$\mathrm{kg}^{-1}\ \mathrm{m}^{-3}$, 1	(normalised) droplet number concentration per mass interval
$g_m, g_{\ln r}$	m^{-3}, $\mathrm{kg\ m}^{-3}$	droplet mass concentration per mass interval/per logarithmic radius interval
$m,\ m'$	kg	mass of a single real droplet
m_{bb}	kg	bin boundaries of the bin grid
~~$\bar{m} = \lambda_1/\lambda_0 = \mathcal{N}/\mathcal{M}$~~ $\bar{m} = \lambda_1/\lambda_0 = \mathcal{M}/\mathcal{N}$	kg	mean mass of all droplets
$n_{bin,l}$	1	droplet number in bin l
$r,\ r'$	m	droplet radius
r_{lb}	m	threshold radius in $\nu_{random,lb}$-init
$r_{critmin}$	m	lower cut-off radius in singleSIP-init
w_{sed}	$\mathrm{m\,s}^{-1}$	sedimentation velocity
$DNC = \lambda_0$	m^{-3}	droplet number concentration
E_c	1	collection/aggregation efficiency
K	$\mathrm{m}^3\ \mathrm{s}^{-1}$	collection/aggregation kernel
$LWC = \lambda_1$	$\mathrm{kg\,m}^{-3}$	droplet mass concentration, liquid water content
$M_{bin,l}$	kg	total droplet mass in bin l
N_{SIP}	1	number of SIPs
N_{BIN}	1	number of bins
$\alpha_{low}, \alpha_{med}, \alpha_{high}$	1	parameters of the ν_{random}-init method.
Δt	s	time step
ΔV	m^3	grid box volume
η	1	parameter in RMA algorithm and singleSIP-init method
κ	1	number of bins per mass decade
λ_k	$\mathrm{kg}^k\ \mathrm{m}^{-3}$	moments of the order k
μ	kg	single droplet mass of a SIP
$\nu_{critmax}$	1	maximum number of droplets represented by a SIP
$\nu_{critmin}$	1	minimum number of droplets represented by a SIP
ν	1	number of droplets represented by a SIP
ξ	1	splitting parameter of AON algorithm
$\chi = \mu\,\nu,\ \tilde{\chi} = \chi/\mathcal{M}$	kg, 1	total droplet mass of a SIP
$\mathcal{N} = \lambda_0 \Delta V$	1	total droplet number
$\mathcal{M} = \lambda_1 \Delta V$	kg	total droplet mass
$\mathcal{Z} = \lambda_2\,\Delta V$	kg^2	second moment of droplet mass distribution (radar reflectivity)

Table 2. List of abbreviations.

AON	All-Or-Nothing algorithm	AIM	Average Impact algorithm
DSD	droplet size distribution	LCM	Lagrangian Cloud Model
PDF	probability density function	RMA	Remapping algorithm
OTF	Update on the fly	RedLim	Reduction Limiter
SIP	simulation particle		

study), the probability density function (PDF) reads as

$$f_m(m) = \frac{\mathcal{N}}{\Delta V \bar{m}} \exp\left(-\frac{m}{\bar{m}}\right), \tag{7}$$

the moments are given analytically by

$$\lambda_{k,anal}(t) = (k-1)!\, \mathcal{N}\, \bar{m}^k / \Delta V, \tag{8}$$

where $k!$ is the ~~faculty~~ factorial of k and $\bar{m} = \mathcal{M}/\mathcal{N}$ the mean mass (Rade and Westergren, 2000).

Throughout this study, the initial parameters of the droplet size distribution (DSD) are $DNC_0 = 2.97 \times 10^8$ m^{-3} and $LWC_0 = 10^{-3}$ kg m^{-3} (implying a mean radius of $9.3\,\mu$m) as in Wang et al. (2007). The higher moments are $\lambda_{2,anal} = 6.74{\times}10^{-15}$ kg^{2}m^{-3} and $\lambda_{3,anal} = 6.81{\times}10^{-26}$ kg^{3}m^{-3}.

2.1 Initialisation

In our test cases, all microphysical processes except collection are neglected and an exponential DSD is initialised. In the results section, we will demonstrate that the outcome of the various collection algorithms critically depends on how this initial, analytically defined, continuous DSD is translated into a discrete ensemble of SIPs. Hence, the SIP initialisation is described in some detail.

2.1.1 SingleSIP-init and MultiSIP-init

First, the mass distribution is discretized on a logarithmic scale. The boundaries of bin l are given by $m_{bb,l} = m_{low} 10^{l/\kappa}$ and $m_{bb,l+1}$, where m_{low} is the minimum droplet mass considered. The bin centre is computed using the arithmetic mean $\bar{m}_{bb,l} = 0.5\,(m_{bb,l+1} + m_{bb,l})$. The bin size is $\Delta m_{bb,l} = (m_{bb,l+1} - m_{bb,l})$. The mass increases tenfold every κ bins. Several previous studies used the parameter s with $m_{bb,l+1}/m_{bb,l} = 2^{1/s}$ to characterise the bin resolution. The parameters s and κ are related via $s = \kappa \log_{10}(2) \approx 0.3\,\kappa$.

For each bin, the droplet number is approximated by $\nu_b = f_m(\bar{m}_{bb,l})\,\Delta m_{bb,l} \Delta V$ and one SIP with weighting factor $\nu_{sim} = \nu_b$ and droplet mass $\mu_{sim} = \bar{m}_{bb,l}$ is created, if ν_b is greater than a lower cut-off threshold $\nu_{critmin}$. No SIP is created ~~,~~if $\nu_b < \nu_{critmin}$. Moreover, no SIPs are created from bins with radius $r < r_{critmin}$. We will refer to this as deterministic singleSIP-init. In its probabilistic version, the mass μ_{sim} is randomly chosen within each bin l and $\nu_{sim} = f_m(\mu_{sim})\,\Delta m_{bb,l} \Delta V$ is adapted accordingly. By default, $r_{critmin} = 0.6\,\mu$m and $\nu_{critmin} = \eta \times \nu_{max}$, which is determined

from the maximal weighting factor within the entire SIP ensemble ν_{max} and the prescribed ratio of the minimal to the maximal weighting factor $\eta = 10^{-9}$. For larger $r_{critmin}$ it is advantageous to initialise one additional "residual" SIP that contains the sum of all neglected contributions.

Following Unterstrasser and Sölch (2014, see their Appendix A), we introduce the multiSIP-init technique. It is similar to the singleSIP-init technique, except that we additionally introduce an upper threshold $\nu_{critmax}$. If $\nu_b > \nu_{critmax}$ is fulfilled for a specific bin, then this bin is divided into $\kappa_{sub} = \lceil \nu_b / \nu_{critmax} \rceil$ sub-bins and a SIP is created for each sub-bin. The multiSIP-init technique gives a good trade-off between resolving low concentrations at the DSD tails and high concentrations of the most abundant droplet masses. By default, $\nu_{critmax} = 0.1\,\nu_{max}$.

So far, we introduced initialisation techniques with a strict lower threshold $\nu_{critmin}$ with no SIPs created in bins with $\nu_b < \nu_{critmin}$. We can relax this condition by introducing—what we call—a *weak* threshold. This means, that in such low contribution bin (with $\nu_b < \nu_{critmin}$) we create a SIP with the probability $p_{create} = \nu_b / \nu_{critmin}$ and weighting factor $\nu_{sim} = \nu_{critmin}$. Having many realisations of initial SIP ensembles, the expectation value of the droplet number represented by such SIPs, $\nu_{critmin} \cdot p_{create} + 0 \cdot (1 - p_{create})$, equals the analytically prescribed value ν_b. Using a strict threshold the droplet number would be simply 0 in those low contribution bins. In a related problem, such a probabilistic approach has been shown to strongly leverage the sensitivity of ice crystal nucleation on the numerical parameter $\nu_{critmin}$. This led to a substantial reduction of the number of SIPs that are required for converging simulation results (Unterstrasser and Sölch, 2014).

Using the probabilistic version and a weak lower threshold is particularly important if different realisations of SIP ensembles of the same analytic DSD should be created. The number of SIPs N_{SIP} depends on ~~κ, $\nu_{critmin}$, $\nu_{critmax}$~~ κ, $\nu_{critmin}$, $\nu_{critmax}$ and the parameters of the prescribed distribution.

Moreover, the singleSIP-init is used in a hybrid version, where different κ-values are used in specified radius ranges.

Table 3 lists the resulting number of SIPs for the range of κ-values used in simulations with the probabilistic singleSIP-init and variants of it.

2.1.2 ν_{const}-init and ν_{draw}-init

The accumulated PDF $F(m)$ is given by $\int_0^m \tilde{f}_m(m')dm'$ with the normalised PDF $\tilde{f}_m = f_m / \lambda_0$. First, the size N_{SIP} of the SIP ensemble that should approximate the initial DSD is specified. For each SIP, its mass μ_i is reasonably picked by

$$\mu_i = F^{-1}(\text{rand}()), \tag{9}$$

where rand() generates uniformly distributed random numbers $\in [0, 1]$. In case of the ν_{const}-init, the weighting factors of all SIPs are equally $\nu_i = \nu_{const} = \mathcal{N} / N_{SIP}$. This init method reproduces SIP ensembles similar to the ones in Shima et al. (2009) or Hoffmann et al. (2015). As a variety of

Table 3. Number of SIPs for the probabilistic singleSIP-init method (and variants like the MultiSIP-init) as a function of κ. The given values are averages over 50 realisations and rounded to the nearest integer. SUPP refers to the supplement of this paper.

	κ								
	5	10	20	40	60	100	200	400	
init method	N_{SIP}								Fig.
singleSIP	24	49	98	197	296	494	988	1976	10, 12, 14, 18
multiSIP		256	517	775	1295				19
singleSIP; $r_{critmin} = 1.6\,\mu$m		74	149	223	372				19
singleSIP; $r_{critmin} = 3.0\,\mu$m		58	116	173	228				SUPP
singleSIP; $r_{critmin} = 5.0\,\mu$m		45	89	113	221				SUPP
singleSIP; $t_{init} = 10$ min		58	114	227	339	565			SUPP
singleSIP; $t_{init} = 20$ min		72	142	284	426	709			21
singleSIP; $t_{init} = 30$ min		89	176	352	527	878			SUPP

the ν_{const}-init method, the weighting factors ν_i in the ν_{draw}-init method are simply perturbed by $\nu_i = 2\,\text{rand}()\,\nu_{const}$.

For the case of an exponential distribution, the following holds for the SIPs $i = 1, N_{SIP}$:

$$\mu_i = -\bar{m}\log(\text{rand}()). \tag{10}$$

In the literature, this approach is known as inverse transform sampling. A proof of correctness can be found in classical textbooks, e.g. Devroye (1986, their section II.2).

2.1.3 ν_{random}-init

The third approach allows specifing the spectrum of weighting factors that should be covered by the SIP ensemble. Similar to the ν_{draw}-init method, the weighting factors are randomly determined. Whereas the latter method produced a SIP ensemble with weighting factors uniformly distributed in ν, the ν_{random}-init produces weighting factors uniformly distributed in $\log(\nu)$ and covering the range $[\mathcal{N}\,10^{\alpha_{low}}, \mathcal{N}\,10^{\alpha_{high}}]$. The eventual number of SIPs depends most sensitively on the parameter α_{high}, which controls how big the portion of a single SIP can be.

SIPs with weighting factors $\nu_i = \mathcal{N}\,10^{(\alpha_{low}+(\alpha_{high}-\alpha_{low})\cdot\text{rand}())}$ are created, until $\sum_{j=1}^{N_{SIP}}\nu_j$ exceeds $\mathcal{N}$. The weighting factor of the last SIP is corrected such that $\sum_{j=1}^{N_{SIP}}\nu_j = \mathcal{N}$ holds. Now the mass μ_i of each SIP is determined by the following technique: The first SIP represents the smallest droplets and covers the mass interval $[0, m_1]$, whereas the last SIP represents the largest droplets in the interval $[m_{N_{SIP}-1}, \infty]$. The SIPs i in between cover the adjacent mass intervals $[m_{i-1}, m_i]$. The boundaries are implicitly determined by $\int_0^{m_i} f_m(m')dm'\,\Delta V = \sum_{j=1}^{i}\nu_j$. The total mass contained in each SIP is given by $\chi_i = \int_{m_{i-1}}^{m_i} f_m(m')m'dm'\,\Delta V$ and the single droplet mass by $\mu_i = \chi_i/\nu_i$.

For the case of an exponential distribution, the following holds for the interval boundaries and the SIPs $i = 1, N_{SIP}$:

$$m_i = -\bar{m} \log \left(\mathcal{N} - \sum_{j=0}^{i} \nu_j \frac{\mathcal{N} - \sum_{j=0}^{i} \nu_j}{\mathcal{N}} \right) \tag{11}$$

and

$$\mu_i = \left(\frac{m_{i-1} - \bar{m}}{\exp(m_{i-1}/\bar{m})} - \frac{m_i - \bar{m}}{\exp(m_i/\bar{m})} \right) \frac{\mathcal{N}}{\nu_i}. \tag{12}$$

The above formulas, which involve several differences of similarly valued terms, must be carefully implemented such that numerical cancellation errors are kept tolerable.

Experimenting with the SIP-init procedure, several optimisations have been incorporated. First, the ν-spectrum is split into two intervals $[\mathcal{N}\,10^{\alpha_{low}}, \mathcal{N}\,10^{\alpha_{med}}]$ and $[\mathcal{N}\,10^{\alpha_{med}}, \mathcal{N}\,10^{\alpha_{high}}]$. We alternately pick random values from the two intervals. Without this correction, it happened that several consecutive SIPs with small weights and hence nearly identical droplet masses are created, which increases the SIP number without any benefits.

Going through the list of SIPs, the droplet masses increase and hence the individual SIPs contain gradually increasing fractions of the total grid box mass. This can lead to a rather coarse representation of the right tail of the DSD. Two options to improve this have been implemented. In the $\nu_{random,rs}$-option, the ν_i-values are reduced by some factor, that increases, as $\sum_{j=1}^{i} \nu_j$ approaches $\mathcal{N}$. In the $\nu_{random,lb}$-option, ν-values are randomly picked up to a certain radius threshold r_{lb}. Above this threshold, SIPs are created with the singleSIP-method ~~used on a linear bin~~with linearly spaced bins.

2.1.4 Comparison

Figure 1 shows the weighting factors and other properties of the initial SIP ensemble, which may affect the performance of the algorithms. Each column shows one class of initialisation techniques. For a certain realisation, the first row shows the weighting factors ν_i of all SIPs as a function of their represented droplet radius r_i. Each dot shows the (ν_i, r_i)-pair of one SIP. For the singleSIP-init, the dots are ~~uniformly~~ homogeneously distributed along the horizontal axis, as one SIP is created from each bin (with exponentially increasing bin sizes). The according ν-values relate directly to the prescribed DSD. The higher $f_m \Delta m$, the more droplets are represented in a SIP. No SIPs smaller than $r_{critmin} = 0.6\,\mu\mathrm{m}$ are initialised and the ν-values range over nine orders of magnitude consistent with $\eta = 10^{-9}$. The MultiSIP-init introduces an upper bound of ~~$\nu_{critmax} = 2 \cdot 10^6$~~ $\nu_{critmax} = 2.6 \cdot 10^6$ for ν. This threshold is effective over a certain radius range where the SIPs~~have lower ν-values~~, compared to the singleSIP-init, have lower ν-values and are also more densely distributed along the horizontal axis. For the ν_{const}-init, all SIPs use $\nu = \nu_{const}$, whereas for the

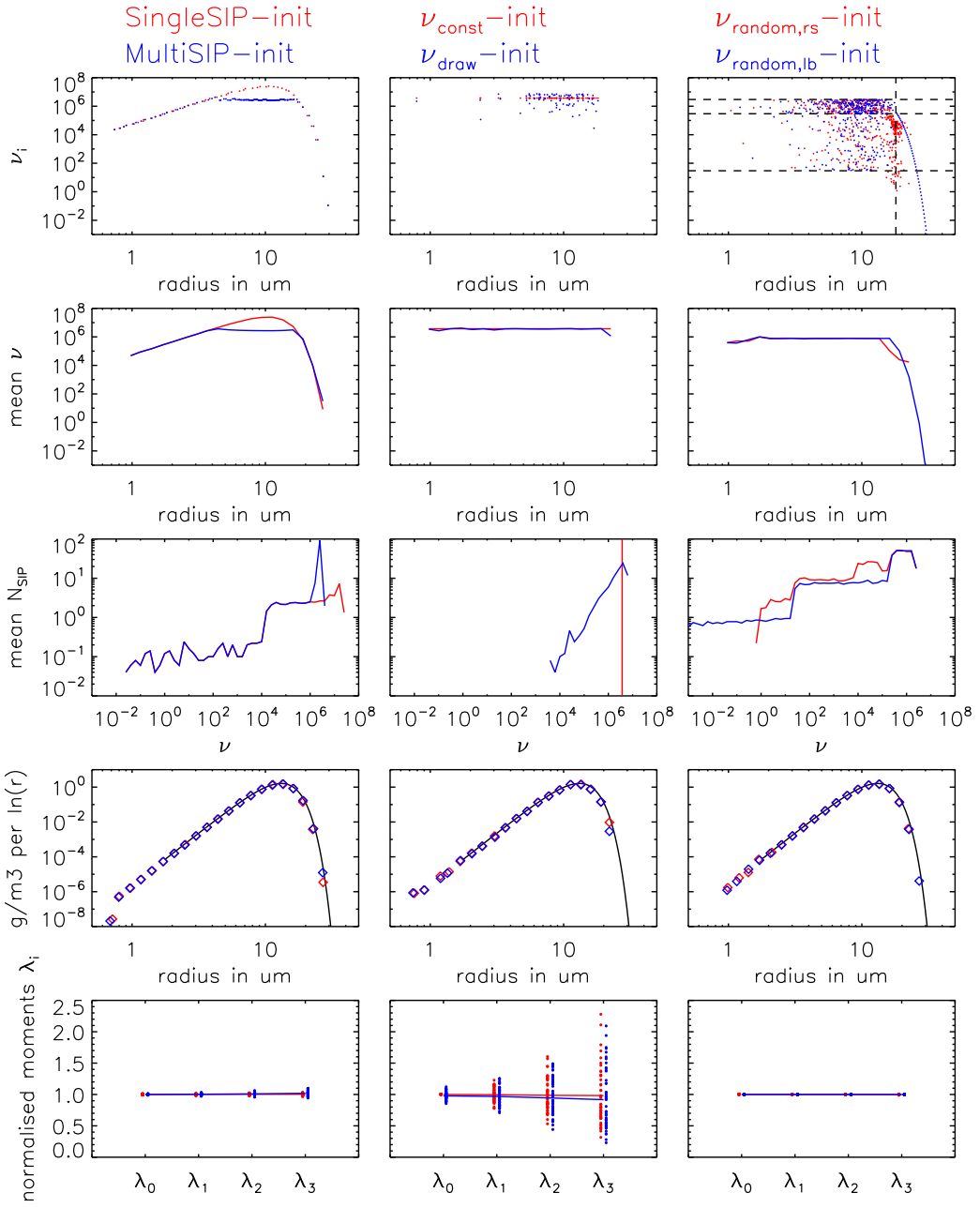

Figure 1. Characteristics of the various SIP initialisation methods (as given on top of each panel): Weighting factors $\nu_i(r_i)$ of an initial SIP ensemble, the mean weighting factors $\bar{\nu}(r)$, the occurrence frequency of the ν_i-values and the resulting mass density distributions $g_{\ln r}$ are displayed (Row 1 to 4). Row 1 displays data of a single realisation, whereas rows 2 to 4 show averages over 50 SIP ensembles. The bottom row shows the moments λ_0, λ_1, λ_2 and ~~λ_2~~ λ_3 normalised by the respective analytical value. Every symbol depicts the value of a single realisation. The nearly horizontal line connects the mean values over all realisations. In the displayed examples, ~~$\kappa = 20$~~ $\kappa = 10$ in the singleSIP-init, ~~$\kappa = 20$, $\nu_{critmax} \approx 2 \cdot 10^6$~~ $\kappa = 10$, $\nu_{critmax} \approx 2.6 \cdot 10^6$ in the multiSIP-init, $N_{SIP} = 80$ in the ν_{const}, ν_{draw}-init and ~~$(\alpha_{high}, \alpha_{med}, \alpha_{low}) = (10^{-2}, 10^{-3}, 10^{-7})$~~ $(\alpha_{high}, \alpha_{med}, \alpha_{low}) = (-2, -3, -7)$ in the ν_{random}-inits~~(see black bars in~~ . In top right panel~~). The~~ , the dashed horizontal lines indicate the values of $\mathcal{N}\,10^{\alpha_{low}}$, $\mathcal{N}\,10^{\alpha_{med}}$ and $\mathcal{N}\,10^{\alpha_{high}}$ and the dashed vertical ~~bar depicts~~ line the threshold radius ~~r_{lb}~~.

ν_{draw}-init the ν-values scatter around this value. For ν_{const} and ν_{draw}, the ν-values are chosen independently of the given DSD contrary to the latter techniques. However, for both techniques, the density of the dots along the r-axis is correlated to $f_m \Delta m$.

The ν_{random}-init technique randomly picks ν-values which are distributed over a larger range compared to the ν_{draw}-init. In fact, they are uniformly distributed in $\log(\nu)$. The range of possible ν-values can be adjusted and is chosen similar to the singleSIP/multiSIP by setting $\alpha_{high} = -2, \alpha_{med} = -3$ and $\alpha_{low} = -7$~~. One possible advantage~~ , which is the default in all simulations presented here. The present method is more flexible compared to the singleSIP-approach ~~could be that~~ as the occurrence of certain ν-values is not limited to a certain radius range. In the singleSIP-init, the smallest ν-values occur only at the left and right tail of the DSD, whereas in the ν_{random}-approach the smallest ν-values (down to $\mathcal{N}\, 10^{\alpha_{low}}$) can appear over the whole radius range. The horizontal ~~bars in the plot~~ lines in the top right panel indicate the values of ~~$\alpha_{low}, \alpha_{med}$ and α_{high}~~ $\mathcal{N}\, 10^{\alpha_{low}}$, $\mathcal{N}\, 10^{\alpha_{med}}$ and $\mathcal{N}\, 10^{\alpha_{high}}$ and the vertical ~~bar~~ line the threshold radius r_{lb}.

The second row shows average ν-value of all SIPs in a certain size bin. All init techniques are probabilistic and the average is taken over 50 independent realisations of SIP ensembles. Not surprisingly, the average ν of the ν_{draw}-method is identical to ν_{const}. Moreover, also for the ν_{random}-init the average ν-value is constant over a large radius range. Only in the right tail, the ν-values drop as intended. The third row shows the occurrence frequency of weighting factors.

To display DSDs represented by a SIP ensemble, a SIP ensemble must be converted back into a bin representation. For this, we establish a grid with resolution $\kappa_{plot} = 4$, count each SIP in its respective bin, i.e. SIP i with $m_{bb,l} < \mu_i \leq m_{bb,l+1}$ contributes to bin l via $M_{bin,l} = M_{bin,l} + \mu_i \times \nu_i$ and $n_{bin,l} = n_{bin,l} + \nu_i$. We note that all displayed DSDs in this study will use $\kappa = 4$, irrespective of the κ-value chosen in the initialisation. The fourth row shows such DSDs, again as an average over 50 SIP ensemble realisations. We find that any init technique is, in general, successful in producing a meaningful SIP ensemble as the "back"-translated DSD matches the originally prescribed DSD (black). Hence, the moments $\lambda_{k,SIP}$ match the analytical values $\lambda_{k,anal}$ for $0 \leq k \leq 3$, as shown in the fifth row. Nevertheless for the ν_{const}- and ν_{draw}-init, the spread between individual realisations can be large and they deviate substantially from the analytical reference. The singleSIP/~~multiSIP~~ multiSIP-init and ν_{random}-init, on the other hand, guarantee that each individual realisation is fairly close to the reference. In the results section, the presented simulations mostly use the probabilistic singleSIP-initialisation. Table 3 shows lists the number of SIPs for several init methods and parameter configurations. The right most column indicates in which figure the simulations using the specific init method are displayed.

2.2 Description of Hypothetical algorithm

First, we present a hypothetical algorithm for the treatment of collection/aggregation in an LCM, which would probably yield excellent results. However, it is prohibitively expensive in terms of

computing power and memory, as N_{SIP} increases drastically over time until the state is reached where each SIP represents exactly one real droplet. Nevertheless, the presentation of this algorithm is useful for introducing several concepts which will partly occur in the subsequently described "real-world" algorithms.

Whereas condensation/deposition and sedimentation may be computed using interpolated quantities which implicitly assumes that all droplets of a ~~SIPs~~ SIP are located at the same point, the numerical treatment of collection usually assumes that the droplets of a SIP are spatially uniformly distributed, i.e. well-mixed within the grid box. An approach, where the vertical SIP position is retained in the collection algorithm and larger droplets overtaking smaller droplets is explicitly modelled, is described in Sölch and Kärcher (2010) ~~, is~~ and not treated here.

Following Gillespie (1972) and Shima et al. (2009), the probability P_{ij} that one droplet with mass m_i collides with one droplet with mass m_j inside a small volume δV within a short time interval δt is given by:

$$P_{ij} = K_{ij}\,\delta t\,\delta V^{-1}, \tag{13}$$

where $K_{ij} = K(m_i, m_j)$.

For SIPs i and j containing ν_i and ν_j real droplets in a grid box with volume ΔV, on average $\nu_{coll} = P_{ij}\,\nu_i\,\nu_j$ collections between droplets from SIP i and SIP j occur. The average rate of such $i-j$-collections ($i \neq j$) to occur is:

$$\frac{\partial \nu_{coll}(i,j)}{\partial t} = \nu_i\,K_{ij}\,\nu_j \Delta V^{-1} =: \nu_i o_{ij} =: O_{ij}. \tag{14}$$

So-called self-collections, collisions of the droplets belonging to the same SIP ($i = j$), are described by:

$$\frac{\partial \nu_{coll}(i,i)}{\partial t} = 2 \cdot \left(\frac{\nu_i}{2}\,K_{ii}\,\frac{\nu_i}{2}\Delta V^{-1}\right) = \frac{1}{2}\nu_i\,K_{ii}\,\nu_i \Delta V^{-1} =: \nu_i o_{ii} =: O_{ii}, \tag{15}$$

assuming that the SIP is split into two portions, each containing one half of the droplets of the original SIP. The factor of 2 originates from the collections of each half, which have to be added to gain the total number of self-collections for SIP i. Accordingly, the diagonal elements of the matrices o_{ij} and O_{ij} differ from the off-diagonal elements by an additional factor of 0.5. In terms of concentrations (represented by SIPs in a grid box with volume ΔV), we can write

$$\frac{\partial n_{coll}(i,j)}{\partial t} = K_{ij}\,n_i\,n_j \tag{16}$$

for collections between different SIPs and

$$\frac{\partial n_{coll}(i,i)}{\partial t} = \frac{1}{2}\,K_{ii}\,n_i^2 \tag{17}$$

for self-collections.

In the hypothetical algorithm, the weighting factor of SIP i is reduced due to collections with all other SIPs and self-collections and reads as

$$\frac{\partial \nu_i}{\partial t} = -\sum_{j=1}^{N_{SIP}} \frac{\partial \nu_{coll}(i,j)}{\partial t} = -\sum_{j=1}^{N_{SIP}} O_{ij}. \tag{18}$$

The droplet mass μ_i in SIP i is unchanged.

For each $i-j$-combination, a new SIP k is generated:

$$\frac{\partial \nu_k}{\partial t} = O_{ij} \quad \text{and} \quad \mu_k = \mu_i + \mu_j \tag{19}$$

To avoid double counting only combinations with $i \geq j$ are considered.

The rate equations for the weighting factors can be numerically solved by a simple Euler forward step. The weighting factor of existing SIPs is reduced by

$$\nu_i^{\Delta} := \left(\sum_{j=1}^{N_{SIP}} O_{ij} \right) \Delta t \tag{20}$$

leading to

$$\nu_i^* = \nu_i - \nu_i^{\Delta}, \tag{21}$$

or, equivalently,

$$\nu_i^* = \nu_i \left(1 - \Delta t \sum_{j=1}^{N_{SIP}} o_{ij} \right). \tag{22}$$

For new SIPs k we have

$$\nu_k = 0 + O_{ij} \cdot \Delta t. \tag{23}$$

Per construction the algorithm is mass-conserving subject to rounding errors.

In each time step, $N_{SIP,add} = N_{SIP}\,(N_{SIP}-1)/2$ new SIPs are produced and the new number of SIPs is ${N_{SIP}}^* = N_{SIP} + N_{SIP,add}$. After nt time steps, the number of SIPs would be of order $(N_{SIP,0})^{nt}$ which is not feasible.

In the following subsections, algorithms are presented that include various approaches to keep the number of SIPs in an acceptable range.

In the following the various algorithms are described and pseudo-code of the implementations is given. For the sake of readability, the pseudo-code examples show easy-to-understand implementations. The actual codes of the algorithms are, however, optimised in terms of computational efficiency. The style conventions for the pseudo-code examples are as follows: Commands of the algorithms are written in upright font with keywords in boldface. Comments appear in italic font (explanations are embraced by {} and headings of code blocks are in boldface).

Algorithm 1 Pseudo-code of the Remapping algorithm (RMA); style conventions are explained at the end of Section 2.2

1: ***INIT BLOCK***
2: Given: Ensemble of SIPs; Specify: $\kappa, \eta, \Delta t$
3: **for** $l = 1$ **to** l_{max} **do** *{Create temporary bin}*
4: $m_{bin,l} = m_{bin,low} 10^{l/\kappa}$
5: **end for**
6: ***TIME ITERATION***
7: **while** t<Tsim **do**
8: ***LOSS BLOCK*** *{Compute reduced bin contribution of existing SIPs}*
9: **for** $i = 1$ **to** N_{SIP} **do**
10: Calculate ν_i^* according to Eq. 22
11: Select bin l with $m_{bb,l} < \mu_i \leq m_{bb,l+1}$
12: $n_{bin,l} = n_{bin,l} + \nu_i^*$
13: $M_{bin,l} = M_{bin,l} + \nu_i^* \cdot \mu_i$
14: **end for**
15: ***GAIN BLOCK*** *{Compute bin contribution of coalescing droplets}*
16: $k = 0$
17: **for all** $i < j \leq N_{SIP}$ **do**
18: ~~k++~~ $k = k + 1$
19: Compute ν_k according to Eq. 23
20: $\mu_k = \mu_i + \mu_j$
21: Select bin l with $m_{bb,l} < \mu_k \leq m_{bb,l+1}$
22: $n_{bin,l} = n_{bin,l} + \nu_k$
23: $M_{bin,l} = M_{bin,l} + \nu_k \cdot \mu_k$
24: **end for**
25: ***CREATE BLOCK*** *{Replace SIPs}*
26: Delete all SIPs
27: $i = 0$
28: **for all** l with $M_{bin,l} > M_{critmin} = \eta \lambda_1$ **do** *{use $M_{critmin}$ as a weak threshold value}*
29: ~~i++~~ $i = i + 1$
30: Generate SIP i with $\nu_i^{new} = n_{bin,l}$ and ~~$\mu_i = M_{bin,l}/\nu_{bin,l}$~~ $\mu_i = M_{bin,l}/n_{bin,l}$
31: **end for**
32: ~~$N_{SIP} = ii$~~ $N_{SIP} = i$
33: $t = t + \Delta t$
34: **end while**
35: ***EXTENSIONS***
36: *Self-collections for a kernel with $K(m,m) \neq 0$ can be easily ~~incorporating~~ incorporated in the algorithm by changing the condition in line 17 to $i \leq j \leq N_{SIP}$.*

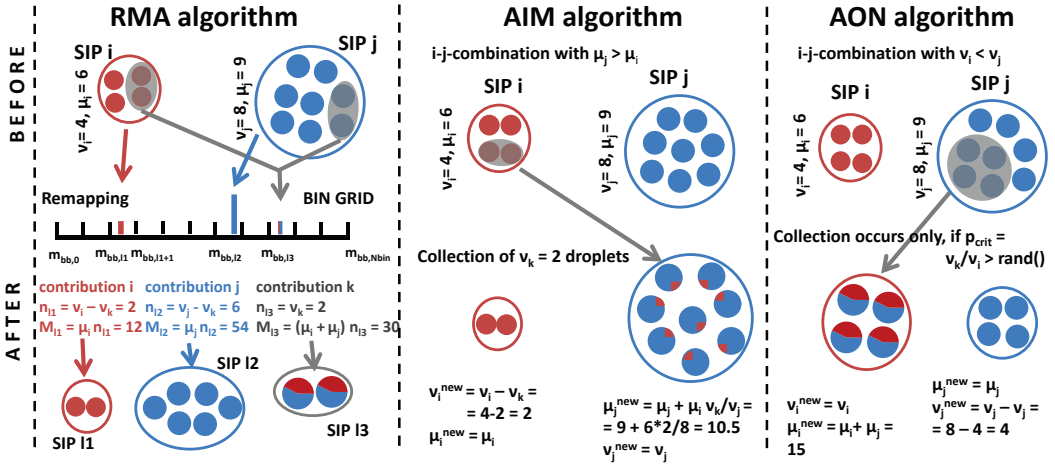

Figure 2. Treatment of a collection between two SIPs in the Remapping Algorithm (RMA), Average Impact Algorithm (AIM) and All-Or-Nothing Algorithm (AON).

2.3 Description of the Remapping (RMA) algorithm

First, the remapping algorithm is described as its concept follows closely the hypothetical algorithm introduced in the latter section. The RMA algorithm is based on ideas of Andrejczuk et al. (2010). We call their approach 'remapping algorithm' as N_{SIP} is kept reasonably low by switching between a SIP representation and a bin representation in every time step. A temporary bin grid with a pre-defined κ is established which stores the total number $n_{bin,*}$ and total mass $M_{bin,*}$ of all contributions belonging to a specific bin. The bin boundaries are given by $m_{bb,*}$.

Instead of creating a new SIP k (with number ν_k obtained by Eq. 19 and mass $\mu_k = \mu_i + \mu_j$) from each $i-j$-combination, the according contribution is stored on a temporary bin grid. More explicitly, this means that the droplet number $n_{bin,l}$ of bin l with $m_{bb,l} < \mu_k \leq m_{bb,l+1}$ is increased by ν_k. Similarly, the total mass $M_{bin,l}$ of that bin is increased by $\mu_k \nu_k$. Similarly, the reduced contributions ν_i^* from the existing SIPs with droplet mass μ_i are added to their respective bins.

Figure 2 illustrates how a collection process between two SIPs is treated in RMA. In this example, $\nu_k = 2$ droplets are produced by collection which have a droplet mass of $\mu_k = \mu_i + \mu_j = 15$. Instead of creating a new SIP k (as in the hypothetical algorithm), the contribution k is recorded in the bin grid. The droplet number n in bin $l3$ is increased by $\nu_k = 2$ and the according total mass M_{l3} by $\nu_k \mu_k = 30$. The remaining contribution of SIP i falls into bin $l1$ and n_{l1} and M_{l1} are increased by $\nu_i^* = \nu_i - \nu_k = 2$ and $\mu_i \nu_i^* = 12$, respectively. The operation for SIP j is analogous.

At the end of each time step after treating all possible $i-j$-combinations, a SIP ensemble is created from the bin data with $\nu_i = n_{bin,l}$ and $\mu_i = M_{bin,l}/n_{bin,l}$, which resembles a deterministic singleSIP-init with the resolution κ.

Optionally, a lower threshold $\nu_{min,RMA}$ can be introduced, such that SIP i is created only if $n_{bin,l} > \nu_{min,RMA}$ holds. However, this may destroy the property of mass conservation which can be remedied by the following.

We pick up the concept of a weak threshold introduced earlier and adjust it such that on average the total mass is conserved (instead of total number as before). We introduce the threshold $M_{critmin} = \eta\lambda_1$. ~~E.g.~~ The parameter η is set to ~~10^{-10}~~ 10^{-8}, which implies that each SIP contains at least ~~10^{-10} of~~ a fraction of 10^{-8} of the total mass in a grid box. If $M_{bin,l} > M_{critmin}$, a SIP is created representing $\nu_i = n_{bin,l}$ drops with single mass $\mu_i = M_{bin,l}/n_{bin,l}$. If $M_{bin,l} < M_{critmin}$, a SIP is created with probability $p_{create} = M_{bin,l}/M_{critmin}$. In this case the SIP represents $\nu_i = M_{critmin}/\mu_i$ droplets with single mass $\mu_i = M_{bin,l}/n_{bin,l}$. Pseudo-code of the algorithm is given in algorithm 1.

Time steps typically used in previous collection/aggregation tests are around $\Delta t = 0.1$ to $10\,\mathrm{s}$ depending inter alia on the used kernel. From Eq. 22 follows that the time step in RMA must satisfy

$$\Delta t < \sum_{j=1}^{N_{SIP}} o_{ij}. \tag{24}$$

Otherwise, negative ν-values can occur which would inevitably lead to a crash of the simulation. In mature clouds, the Long and Hall kernel attain large values which required tiny time steps of $10^{-4}\,\mathrm{s}$ and smaller in the first test simulations. To be of any practical relevance, RMA had to be modified in order to be able to run simulations with suitable time steps.

Hence, several extensions to RMA allowing larger time steps are ~~discussed~~ proposed in the following.

1. *Default version:* Use the algorithm as outlined in Algorithm 1 (i.e. do not change anything). Negative ν_i^*-values obtained by Eq. 21 are acceptable, as long as $n_{bin,l}$, from which the SIPs are created at the end of the time iteration, is non-negative for all l. This means that an existing SIP i (which falls into bin l) can lose more droplets (ν_i^Δ) than it actually possesses (ν_i) as long as the gain in bin l (from all suitable SIP combinations) compensates this deficit. We will later see that this approach works well for the Golovin kernel, however fails for the Long and Hall kernel.

2. *Clipping:* Simply ignore bins with negative $n_{bin,l}$ and do not create SIPs from those bins. This approach destroys the property of mass conservation and is not pursued here.

3. *Adaptive time stepping:* Instead of reducing the general time step, only the treatment of SIPs with $\nu_i^* < 0$ is sub-cycled. For each such SIP i, Eq. 21 is iterated $\tilde{\eta}_i$ times with time step $\Delta t_{SIP} = \Delta t/\tilde{\eta}_i$. Note that even though the computation of Eq. 21 and O_{ij} involves the ν-evaluation of all SIPs, only ν_i is updated in the subcycling steps and not the whole system of fully coupled equations is solved for a smaller time step. For sufficiently large $\tilde{\eta}_i$, $\nu_{i,subcycl}^*$ is positive, as $\nu_{i,subcycl}^\Delta < \nu_i$ as desired. Basically, we now assume that all collections involving

SIP i are equally reduced by a factor of $\eta_i = \nu^{\Delta}_{i,subcycl}/\nu^{\Delta}_i$ compared to the default time step. In the GAIN block of the algorithm (as termed in Alg. 1), all computations use the default time step and no sub-cycling is applied. To be consistent with the reduction in the LOSS block, Eq. 23 is replaced by $\nu_k = \eta_i O_{ij} \Delta t$.

4. *Reduction ~~limiter~~Limiter (abbr. as RedLim)* The effect of an adaptively reduced time step can be reached with simpler and cheaper means. We introduce a threshold parameter $0 < \tilde{\gamma} < 1.0$ similar to the approach in Andrejczuk et al. (2012). Again, we focus on SIPs with $\nu^*_i < 0$ and simply set the new weight of SIP i to $\nu^*_{i,RedLim} = \tilde{\gamma}\nu_i$. As above, all contributions involving SIP i have to be re-scaled, now with $\gamma_i = (\nu_i - \nu^*_{i,RedLim})/\nu^{\Delta}_i$.

5. *Update on the fly (abbr. as OTF)* Another option to eliminate negative ν_i-values is to do an "update on the fly". In this case, the algorithm is not separated in a LOSS and GAIN block. Instead, the $i-j$-combinations are processed one after another. After each collection process, as exemplified in Fig. 2, the weighting factors ν_i and ν_j of the two involved SIPs are reduced by ν_k, i.e. the number of droplets that were collected. Subsequent evaluations of Eq. 23 then use updated ν-values. Compared to the default version, it now matters in which order the $i-j$-combinations are processed, e.g. if you deal first with combinations of the smallest SIPs or of the largest SIPs.

2.4 Description of Average Impact (AIM) algorithm

The average impact algorithm by Riechelmann et al. (2012) and further developed in Maronga et al. (2015) predicts the temporal change of the weighting factor, ν_i, and the total mass of all droplets represented by each SIP, $\chi_i = \nu_i \mu_i$. In this algorithm, two fundamental interactions of droplets are considered (see also Fig. 7 in Maronga et al., 2015). First, the coalescence of two SIPs of different size. It is assumed that the larger SIP collects a certain amount of the droplets represented by the smaller SIP, which is then equally distributed among the droplets of the larger SIP. As a consequence, the total mass and the weighting factor of the smaller SIP decrease, while the total mass of the larger SIP increases accordingly. Fig. 2 illustrates how a collection between two SIPs is treated. SIP j is assumed to represent larger droplets than SIP i, i.e. $\mu_j > \mu_i$. As in the RMA example before, we say that $\nu_k = 2$ droplets are collected. Then SIP i loses two droplets to SIP j, i.e. ν_i is reduced by 2 and a mass of $\mu_i\nu_k$ is transferred to SIP j where it is distributed among the existing $\nu_j = 8$ droplets. Unlike to RMA, where droplets with mass $\mu_j + \mu_i = 15$ are produced, AIM predicts a droplet mass of $\mu_j + \mu_i\nu_k/\nu_i = 10.5$ in SIP j. Usually, $\nu_k/\nu_i << 1$ and hence the name "average impact" for this algorithm.

Moreover, same-size collisions are considered in each SIP. This decreases the weighting factor of each SIP but not its total mass. Accordingly, the radius of the SIP increases.

Algorithm 2 Pseudo-code of the average impact algorithm (AIM); style conventions are explained at the end of Section 2.2

1: ***INIT BLOCK + SIP SORTING***
2: Given: Ensemble of SIPs; Specify: Δt
3: ***TIME ITERATION***
4: **while** t<Tsim **do**
5: *{Sort SIPs by droplet mass}*
6: Apply (adaptive) sorting algorithm, such that $\mu_j \geq \mu_i$ for $j > i$
7: *{Compute total mass χ_i of each SIP}*
8: $\chi_i = \nu_i \, \mu_i$
9: **for** $i = 1$ **to** N_{SIP} **do**
10: *{Compute reduction of weighting factor due to number loss to all larger SIPs}*
11: $\nu_i^{new} = \nu_i \left(1 - \Delta t \sum_{j=i+1}^{N_{SIP}} o_{ij}\right)$
12: *{Compute mass transfer; mass gain from all smaller SIPs and mass loss to all larger SIPs}*
13: ~~$\chi_i^{new} = \chi_i - \chi_i \Delta t \sum_{j=i+1}^{N_{SIP}} o_{ij} + \sum_{j=1}^{i-1} \chi_j o_{ij} \Delta t$~~ $\chi_i^{new} = \chi_i + \Delta t \left(\sum_{j=1}^{i-1} \chi_j o_{ij} - \chi_i \sum_{j=i+1}^{N_{SIP}} o_{ij}\right)$
14: **end for**
15: $\nu_i = \nu_i^{new}$
16: $\mu_i = \chi_i^{new} / \nu_i^{new}$
17: $t = t + \Delta t$
18: **end while**
19: ***EXTENSIONS***
20: *{Self-collections for a kernel with $K_{ii} \neq 0$ can be incorporated simply by ~~adding~~ starting the ~~term $-0.5 \, \Delta t \, o_{ii}$ inside the bracket on the r.h.s. of~~ summation in line 11 from $j = i$ (see also Eq. (27) in the text).}*

Both processes are represented in the following two equations which are solved for all colliding SIPs (assuming that $\mu_0 \leq \mu_1 \leq \ldots \leq \mu_{N_{SIP}}$):

$$\frac{\mathrm{d}\nu_i}{\mathrm{d}t} = -K_{ii} \frac{1}{2} \frac{\nu_i \nu_i}{\Delta V} - \sum_{j=i+1}^{N_{SIP}} K_{ij} \nu_i \nu_j \Delta V^{-1} \tag{25}$$

and

$$\frac{\mathrm{d}\chi_i}{\mathrm{d}t} = \sum_{j=1}^{i-1} \mu_j \, K_{ij} \nu_i \nu_j \Delta V^{-1} - \mu_i \sum_{j=i+1}^{N_{SIP}} K_{ij} \nu_i \nu_j \Delta V^{-1}. \tag{26}$$

The first term on the right-hand-side of Eq. 25 describes the decrease of ν due to same-size collections, the second term the decrease of ν due to collection by larger SIPs. The first term on the right-hand-side of Eq. 26 describes the gain in total mass due to collections with smaller SIPs, while the second term describes the loss of total mass due to collection by larger SIPs.

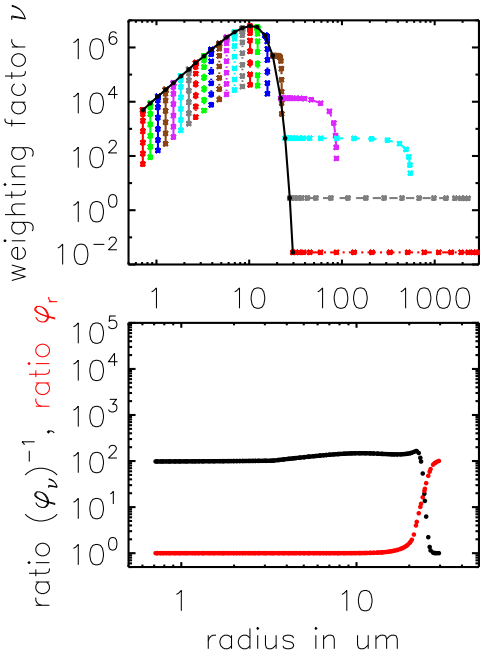

Figure 3. top: (r_i, ν_i)-evolution of selected SIPs for the AIM algorithm. The black line shows the initial distribution. Each coloured line connects the data points that depict the (r_i, ν_i)-pair of an individual SIP every 200 s. bottom: ~~Ratios~~ The ratios φ_r and φ_ν are defined as $r_i(t = 3600\,\mathrm{s})/r_i(t = 0\,\mathrm{s})$ and $\nu_i(t = 3600\,\mathrm{s})/\nu_i(t = 0\,\mathrm{s})$. φ_r (red curve) and ~~$\nu_i(t = 0\,\mathrm{s})/\nu_i(t = 3600\,\mathrm{s})$~~ $(\varphi_\nu)^{-1}$ (black curve) for all SIPs are shown as a function of their initial radius $r_i(t = 0\,\mathrm{s})$.
An example simulation with Long kernel, singleSIP-init, ~~$\Delta t = 10\,\mathrm{s}, \kappa = 40$~~ $\Delta t = 10\,\mathrm{s}$, $\kappa = 40$ and $N_{SIP} = 197$ is displayed.

Using a Euler forward method for time integration the above equations read as:

$$\nu_i^{new} = \nu_i \left(1 - \sum_{j=i+1}^{N_{SIP}} o_{ij} \Delta t - 0.5 o_{ii} t\right) \tag{27}$$

and

$$\chi_i^{new} = \chi_i \left(1 - \sum_{j=i+1}^{N_{SIP}} o_{ij} \Delta t\right) + \sum_{j=1}^{i-1} \chi_j o_{ij} \Delta t. \tag{28}$$

Finally, the single droplet mass μ_i of each SIP is updated: $\mu_i^{new} = \chi_i^{new}/\nu_i^{new}$. Pseudo-code of the algorithm is given in algorithm 2.

Figure ~~2~~ 3 illustrates how the AIM algorithm works for an example simulation with the Long kernel and singleSIP-init. The top panel shows the (r_i, ν_i)-evolution of selected SIPs. The black line shows the initial distribution. Each coloured line connects the data points that depict the (r_i, ν_i)-pair of an individual SIP every 200 s. Clearly, ν_i of any SIP decreases over time, however the decrease is much smaller for the largest SIPs and becomes zero for the largest SIP. The majority of SIPs starting

from the smallest radii show an opposite behaviour as their evolution is dominated by a strong ν_i-decrease at nearly constant r_i. In contrast, the evolution of the two largest SIPs is dominated by a strong r_i-increase for constant ν_i. The SIPs next to the largest SIPs undergo a transition; in the beginning, they primarily grow in size, towards the end the decrease of ν_i is dominant. ~~The bottom panel~~

The ratio φ_r is defined as $r_i(t=3600\mathrm{s})/r_i(t=0\mathrm{s})$ and, analogously, $\varphi_\nu = \nu_i(t=3600\mathrm{s})/\nu_i(t=0\mathrm{s})$. We find $\varphi_r \geq 1$ and $\varphi_\nu \leq 1$. The bottom panel of Figure 3 shows the ratios ~~$r_i(t=3600\mathrm{s})/r_i(t=0\mathrm{s})$~~ φ_r (red curve) and ~~$\nu_i(t=0\mathrm{s})/\nu_i(t=3600\mathrm{s})$~~ $(\varphi_\nu)^{-1}$ (black curve) for all SIPs of the simulation. Both ratios are smooth functions of the initial r_i, which is plotted on the x-axis.

By construction, the number of SIPs remains constant over the course of a simulation. Hence, the number of SIPs per radius or mass interval decreases, when the DSD broadens over time. In our example, the SIP resolution becomes coarser, particularly in the large droplet tail.

Negative values of ν_i^{new} and χ_i^{new} may occur. However, this case never occurred in our manifold tests of the algorithm. The behaviour appears more benign than in RMA. Moreover, we found that the algorithm preserved the initial size-sortedness of the SIP ensemble. However, for an arbitrary kernel function and initial SIP ensemble, this is not guaranteed and we recommend to use adaptive sorting algorithms that benefit from partially pre-sorted data sets (Estivill-Castro and Wood, 1992). Adaptive sorting is also advantageous, when AIM is employed in real world applications, where sedimentation, advection and condensation changes the SIP ensemble in each individual grid box.

2.5 Description of the All-Or-Nothing (AON) algorithm

The All-Or-Nothing (AON) algorithm is based on the ideas of Sölch and Kärcher (2010) and Shima et al. (2009). Fig. 2 illustrates how a collection between two SIPs is treated. SIP i is assumed to represent fewer droplets than SIP j, i.e. $\nu_i < \nu_j$. Each real droplet in SIP i collects one real droplet from SIP j . Hence, SIP i contains $\nu_i = 4$ droplets, now with mass $\mu_i + \mu_j = 15$. SIP j now contains $\nu_j - \nu_i = 8-4 = 4$ droplets with mass $\mu_j = 9$. Following Eq. 23, only $\nu_k = 2$ pairs of droplets would, however, merge in reality. The idea behind this probabilistic AON algorithm is that such a collection event is realised only under certain circumstances in the model, namely such that the expectation values of collection events in the model and in the real world are the same. This is achieved if a collection event occurs with probability ~~$p_{crit} = \nu_k/\nu_i$~~

$$p_{crit} = \nu_k/\nu_i \tag{29}$$

in the model. Then, the average number of collections in the model,

~~$\bar{\nu}_k = p_{crit}\nu_i = (\nu_k/\nu_i)\nu_i,$~~

$$\bar{\nu}_k = p_{crit}\nu_i = (\nu_k/\nu_i)\nu_i, \tag{30}$$

Algorithm 3 Pseudo-code of the all-or-nothing algorithm (AON); style conventions are explained at the end of Section 2.2; rand() generates uniformly distributed random numbers $\in [0,1]$.

1: ***INIT BLOCK***
2: Given: Ensemble of SIPs; Specify: Δt
3: ***TIME ITERATION***
4: **while** t<Tsim **do**
5: *{Check each $i-j$-combination for a possible collection event}*
6: **for all** $i < j \leq N_{SIP}$ **do**
7: Compute ν_k according to Eq. 19
8: $\nu_{new} = \min(\nu_i, \nu_j)$
9: $p_{crit} = \nu_k / \nu_{new}$
10: *{Update SIP properties on the fly}*
11: **if** $p_{crit} > 1$ **then**
12: ***MULTIPLE COLLECTION***
13: *{can occur when ν_i and ν_j differ strongly and be regarded as special case; see text for further explanation}*
14: assume $\nu_i < \nu_j$, otherwise swap i and j in the following lines
15: *{$p_{crit} > 1$ is equivalent to $\nu_k > \nu_i$}*
16: *{transfer ν_k droplets with μ_j from SIP j to SIP i, allow multiple collections in SIP i, i.e. one droplet of SIP i collects more than one droplet of SIP j.}*
17: SIP i collects ν_k droplets from SIP j and distributes them on ν_i droplets: $\mu_i = (\nu_i\,\mu_i + \nu_k\,\mu_j)/\nu_i$
18: SIP j loses ν_k droplets to SIP i: $\nu_j = \nu_j - \nu_k$
19: **else if** p_{crit} >rand() **then**
20: ***RANDOM SINGLE COLLECTION***
21: assume $\nu_i < \nu_j$, otherwise swap i and j in the following lines
22: *{transfer ν_i droplets with μ_j from SIP j to SIP i}*
23: SIP i collects ν_i droplets from SIP j: $\mu_i = \mu_i + \mu_j$
24: SIP j loses ν_i droplets to SIP i: $\nu_j = \nu_j - \nu_i$
25: **end if**
26: **end for**
27: $t = t + \Delta t$
28: **end while**
29: ***EXTENSIONS***
30: *{Self-collections for a kernel with $K(m,m) \neq 0$ can be treated in the following way: }*
31: *{Insert the following loop before line 6 or after line 26.}*
32: **for** $i = 1$ **to** N_{SIP} **do**
33: $p_{crit} = \nu_k / \nu_i$
34: **if** $2\,p_{crit}$ >rand() **then**
35: *{every two (identical) droplets coalesce}*
36: $\nu_i = \nu_i / 2$
37: $\mu_i = 2\,\mu_i$
38: **end if**
39: **end for**

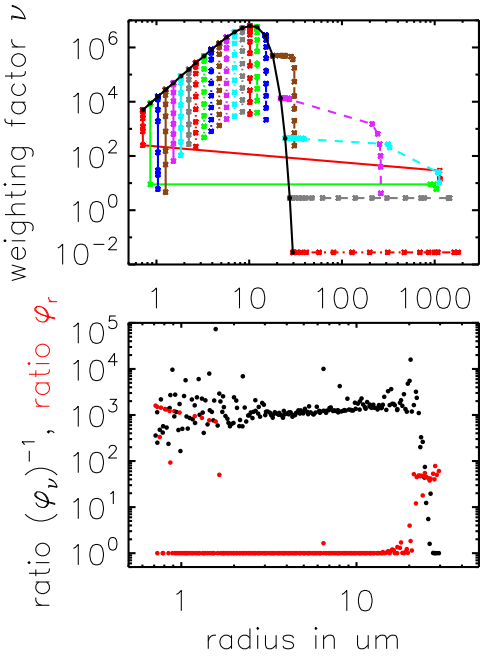

Figure 4. As in Fig. 3, for the AON algorithm.

is equal to ν_k as in the real world. A collection event between two SIPs occurs, if p_{crit} >rand(). The function rand() provides uniformly distributed random numbers $\in [0,1]$.

Noticeably, no operation on ~~the SIPs~~ a specific SIP pair is performed if p_{crit} <rand().

The treatment of the special case $\nu_k/\nu_i > 1$ needs some clarification. This case is regularly encountered when the singleSIP-init is used, where SIPs with large droplets and small ν_i collect small droplets from a SIP with large ν_j. The large difference in droplet masses μ lead to large kernel values and high ν_k with $\nu_i < \nu_k < \nu_j$. By the way, the case of ν_k being even larger than ν_j is not considered, as it occurs only with unrealistically large time steps. If $p_{crit} > 1$, we allow multiple collections, as each droplet in SIP i is allowed to collect more than one droplet from SIP j. In total, SIP i collects ν_k droplets from SIP j and distributes them on ν_i droplets. A total mass of $\nu_k \mu_j$ is transferred from SIP j to SIP i and the droplet mass in SIPs i becomes $\mu_i^{new} = (\nu_i\,\mu_i + \nu_k\,\mu_j)/\nu_i$. The number of droplets in SIP j is reduced by ν_k and $\nu_j^{new} = \nu_j - \nu_k$. Sticking to the example in Fig. 2 and assuming $\nu_k = 5$, each of the $\nu_i = 4$ droplets would collect $\nu_k/\nu_i = 1.25$ droplets. The properties of SIP i and SIP j are then: $\nu_i = 4$, $\mu_i = 17.25$, $\nu_j = 3$ and $\mu_j = 9$.

Another special case appears if both SIPs have the same weighting factor which regularly occurs when the ν_{const}-init is used. After a collection event, SIP j would carry $\nu_j - \nu_i = 0$ droplets, whereas SIP i would still represent ν_i droplets. In this case, half of the droplets from SIP i ~~are moved to~~ coalesce with half of the droplets from SIP j and vice versa. Accordingly, both SIPs carry ~~$\nu_j^{new} = \nu_i^{new} = 0.5\,\nu_i$~~ $\nu_j^{new} = \nu_i^{new} = 0.5 \times \nu_i$ droplets with mass $\mu_i + \mu_j$. Without this correction,

zero-ν SIPs would accumulate over time and reduce the effective number of SIPs causing a poorer sampling. Instead of this equal splitting, one can also assign unequal shares $\xi\,\nu_i$ and $(1-\xi)\nu_i$ to the two SIPs (with ξ being some random number).

Moreover, self-collections can be considered for kernels with $K_{ii} > 0$. If 2 p_{crit} >rand(), self-collections occur between the droplets in a SIP (note the factor 2 due to symmetry reasons). Then every two droplets within a SIP coalesce, implying $\nu_i = \nu_i/2$ and $\mu_i = 2\,\mu_i$.

So far, we explained how a single $i-j$-combination is treated in AON. In every time step, the full algorithm simply checks each $i-j$-combination for a possible collection event. To avoid double-counting only combinations with $i < j$ and self-collections with $i = j$ are considered. Pseudo-code of the algorithm is given in Algorithm 3. The SIP properties are updated on the fly. If a certain SIP is involved in a collection event in the model and changes its properties, all subsequent combinations with this SIP take into account the updated SIP properties. Similar to the update on the fly version of RMA, results may depend on the order in which the $i-j$-combinations are processed.

For most $i-j$-combinations, p_{crit} is small and usually only a limited number of collection events occurs in the model and AON may suffer from an insufficient sampling of the droplet space. Actual collections are a rare event in this algorithm. In our standard setup, $< 1\%$ of all possible collections occur in the model until rain is initiated by very few lucky SIPs (similar to lucky drops, e.g. Kostinski and Shaw (2005)). Indeed, Shima et al. (2009) reported convergence of AON only for tremendously many SIPs (on the order of 10^5 to 10^6 in a box). We will later see that convergence is possible with as few as $O(10^2)$ SIPs, if the SIPs are suitably initialised. Hence, it will be demonstrated that AON is a viable option in 2D/3D cloud simulations, as already implied in Arabas and Shima (2013).

As for AIM in Fig. 3, Fig. 4 (top) shows the (r_i, ν_i)-evolution of selected SIPs for AON. The picture looks more chaotic than for AIM, as each individual SIP has its own independent history due to the probabilistic nature of AON. For the initially smallest SIP, only ν_i changes for most of the time, as only collections occur where the partner SIPs have smaller weighting factors ν. Towards the end, the still very small SIP is at least once involved in a collection with a very large SIP that has a larger ν. Hence, r_i of this SIP increases substantially. In contrast to the smallest SIP, other initially small SIPs i with similar properties are never part of a collection with $\nu_i < \nu_j$. Hence, their radii r_i remain small over the total period and ν_i is the only property that changes. The bottom panel summarises the overall changes in ν_i (black) and r_i (red) for all SIPs of the simulation. Unlike to AIM, where only the initially largest SIPs grow, SIPs from both ends of the spectrum grow in AON. Those SIPs have small ν-values in common and in each collection their mass is updated to $m_i + m_j$. The SIPs with initially large ν-values lie in the radius range $[2\,\mu\mathrm{m}, 15\,\mu\mathrm{m}]$ and keep their initial radii (at least in the singleSIP-init used here). The reductions in ν_i scatter around $\sim 10^3$ for most SIPs and fall off to 1 for the largest SIPs.

For the generation of the random numbers, the well-proven (L'Ecuyer and Simard, 2007) Mersenne Twister algorithm by Matsumoto and Nishimura (1998) is used. AON simulations may be accelerated if random numbers are computed once a priori. However, this requires saving millions of random numbers for every realisation. An AON simulation with 1000 time steps and 200 SIPs~~implying~~, for instance, implies 200×100 ~~combinations, e.g. processes~~ potential collections during one time step and in total $2 \cdot 10^7$ random numbers. Using random numbers with a smaller cycle length deteriorated the simulation results in several tests and is not recommended.

The current implementation differs slightly from the version in Shima et al. (2009). Due to an unfavourable SIP initialisation similar to the ν_{const}-technique, Shima et al. (2009) deal with large N_{SIP}-values in their simulations, where it becomes prohibitive to evaluate all $N_{SIP}(N_{SIP}-1)$ SIP-combinations. Hence, they resort to $\lfloor N_{SIP}/2 \rfloor$ randomly picked $i-j$-combinations, where each SIP appears exactly in one pair (if N_{SIP} is odd, one SIP is ignored). As only a subset of all possible combinations are numerically evaluated, the extent of collisions is underestimated. To compensate for this, the probability p_{crit} is up-scaled with a scaling factor $N_{SIP}(N_{SIP}-1)/(2\,\lfloor N_{SIP}/2 \rfloor)$ to guarantee an expectation value as desired.

Moreover, in Shima's formulation the weighting factors are considered to be integer numbers. In contrast, we use real numbers ν which can even attain values below 1.0. This has several computational advantages: 1. better sampling of the DSD, in particular at the tails, 2. simpler AON implementation with fewer arithmetic and rounding operations, and 3. more flexibility, e.g. SIP splitting with real-valued ξ in the case of identical weighting factors.

Sölch and Kärcher (2010) makes use of the vertical position of the SIPs and explicitly calculates whether or not a larger droplet overtakes a smaller droplet within a time step. This approach will be thoroughly analysed in a follow-up study.

In RMA and AIM, SIPs with negative weights may be generated depending, e.g. on the condition $\Delta t \sum_{j=1}^{N_{SIP}} o_{ij} > 1$ in RMA. ~~In AON,~~ By construction, this cannot happen in AON and the latter condition implies that $\sum_{j=1} p_{crit,ij}$ of SIP i is greater than unity. ~~Hence~~Then, this SIP is likely to be involved in several collections (for j with $p_{crit,ij} < 1$) or is involved in one or several multiple collections (for j with $p_{crit,ij} > 1$).

3 Box model results

In this section, box model simulations of the three algorithms introduced in the latter section are presented, starting with the results of the Remapping (RMA) Algorithm, then those of the Average Impact (AIM) and finally the All-or-Nothing (AON) algorithm. The results of each algorithm are tested for three different collection kernels (Golovin, Long and Hall). As default, probabilistic SIP initialisation methods are used. For each parameter setting, simulations are performed for 50

different realisations. Simulations with the Golovin kernel are compared against the analytical solution given by Golovin (1963). Consistent with many previous studies we choose $b = 1.5\ \mathrm{m^3 kg^{-1} s^{-1}}$.

Simulations with the Long and Hall kernel are compared against high-resolution benchmark simulations obtained by the spectral-bin model approaches of Wang et al. (2007) and Bott (1998). The volume of the box is assumed to be $\Delta V = 1\ \mathrm{m}^3$.

In all simulations, collision/coalescence is the only process considered in order to enable a rigorous evaluation of the algorithms. The evaluation is based on the comparison of mass density distributions, and the temporal development of 0th, 2nd, and 3rd moment of the droplet distributions. The 1st moment is not shown since the mass is conserved in all algorithms per construction. ~~As default, probabilistic SIP initialisation methods are used. For each parameter setting, simulations are performed for 50 different realisations.~~

The supplement (abbreviated as SUPP in the following) contains a large collection of figures that systematically reports all sensitivity tests that have been performed. The behaviour of the second and third moment is similar and the λ_3-evolution is shown only in SUPP. Later it will be mentioned that Hall kernel simulations are not as challenging as Long kernel simulations from a numerical point of view. Hence, simulation with the Hall kernel are only shortly discussed in the manuscript and figures are shown in SUPP.

3.1 Performance of Remapping (RMA) Algorithm

Figure 5 compares DSDs of the RMA algorithm and the analytical reference solution for the Golovin kernel. Each panel displays DSDs from $t = 0$ to $60\,\mathrm{min}$ every $10\,\mathrm{min}$. The upper left panel shows an excellent agreement of RMA with the reference solution and proves at least a correct implementation. Figure 6 compares the temporal evolution of the moments. Moreover, the first row shows the number of SIPs used in RMA. Except for the case with a very coarse grid ($\kappa = 5$) with fewer than 40 SIPs in the end, the ~~RMA results~~ regular RMA results shown in the left column agree perfectly with the reference solution irrespective of the chosen κ (≥ 10) and minimum weak threshold η ranging from 10^{-5} to 10^{-8}. The number of non-zero bins increases as the DSD broadens over time. In the last step of the time iteration, SIPs are created from such bins. Hence, their number increases over time. Using a strict threshold, the total mass is not conserved~~(not shown); The~~ ; the larger η is, the more mass is lost (see SUPP). Hence, using a weak threshold or some other measure (e.g. creation of a residual SIP containing contributions of all neglected bins) to avoid this is highly recommended.

Next, RMA simulations with the Long kernel are discussed. As already mentioned, the default RMA version would require tiny time steps which would rule out RMA from any practical application. Both approaches introduced before, "Update on the fly" (OTF) and "Reduction Limiter" (RedLim), succeed in eliminating negative ν_i-values and in finishing the simulation within a reasonable time. However, the results are not as desired. Fig. 7 shows the DSDs for a simulation with Reduction Limiter $\tilde{\gamma} = 0.1$, weak threshold ~~$\eta = 10^{-8}$ and parameters $\kappa = 60$,~~ $\eta = 10^{-8}$, $\kappa = 20$ and

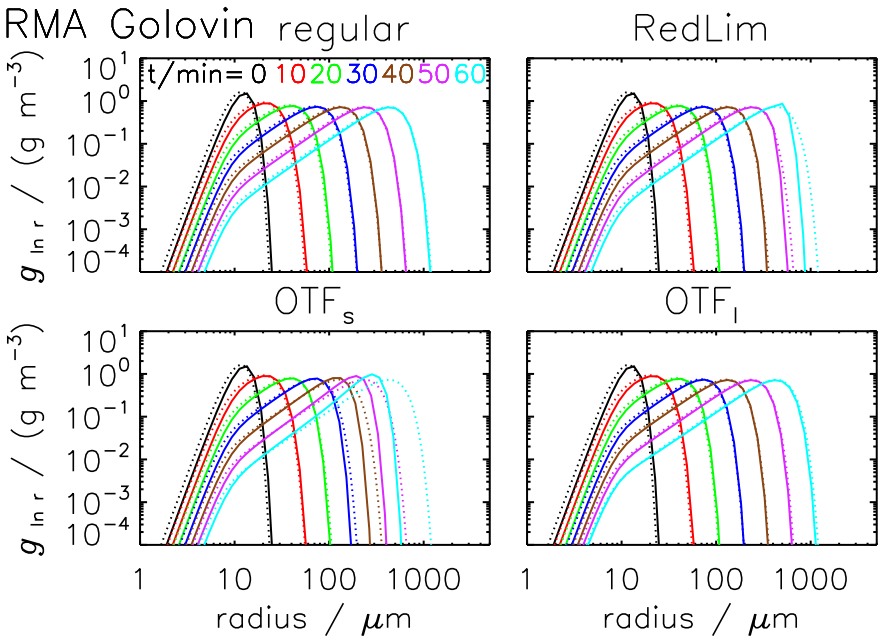

Figure 5. Mass density distributions obtained by the RMA algorithm for the Golovin kernel from $t = 0$ to 60 min every 10 min (from black to cyan, see legend). The ~~solid~~ dotted curves show the reference solution, the ~~dotted~~ solid curves the ~~simulation result of the~~ RMA ~~algorithm~~simulation results (ensemble averages over 50 realisations). The parameter settings are ~~probabilistic~~ singleSIP-init with weak threshold $\eta = 10^{-8}$, $\kappa = 60$ ~~,~~ ~~$\eta = 10^{-8}$~~ and $\Delta t = 1$ s. The following versions of the RMA algorithm are depicted (clockwise from top left~~to right~~): regular version, version with ~~reduction limiter~~Reduction Limiter, version with update on the fly OTF_l and OTF_s (~~start~~ starting with combinations of ~~smallest/~~the largest or smallest droplets, respectively).

$\Delta t = 0.1$ s~~and $\tilde{\gamma} = 0.1$~~. Whereas the algorithm is capable of realistically reducing the number of the smaller droplets, ~~it fails to predict the formation of the rain mode and~~ strong oscillations appear in the intermediate radius range $[100\,\mu\mathrm{m}, 200\,\mu\mathrm{m}]$ ~~.~~ (see right panel). If we average over 50 realisations (as usually, left panel) or use a coarse grain visualisation (as usually with $\kappa_{plot} = 4$, middle panel), the oscillations are smoothed out (or better say masked). Nevertheless, the formation of the rain mode is impeded; probably the mass flux across the problematic radius range is too slow, which is a direct consequence of applying the Reduction Limiter (mostly SIPs in this part of the spectrum obtain negative weights and have to be corrected).

We tested the algorithm ~~with~~ for many parameter settings varying all of the aforementioned parameters, ~~$\Delta t \in [0.1\,\mathrm{s}, 1\,\mathrm{s}], \kappa \in [10, 60], \tilde{\gamma} \in [0, 1]$ and $\eta \in [10^{-10}, 10^{-5}]$. Unfortunately, spurious oscillations occur in most cases. Integrating over the whole mass spectrum, those oscillations do not average out and , not surprisingly, the moments do~~ $\Delta t \in [0.01\,\mathrm{s}, 1\,\mathrm{s}], \kappa \in [5, 100], \tilde{\gamma} \in [0, 1]$ and $\eta \in [10^{-15}, 10^{-5}]$. Figure 8 shows the evolution of moment 0 and 2 for various Δt-values (at $\kappa = 10$, left column) and κ-values (at $\Delta t = 0.1$ s right column). Obviously, the simulation results are nearly

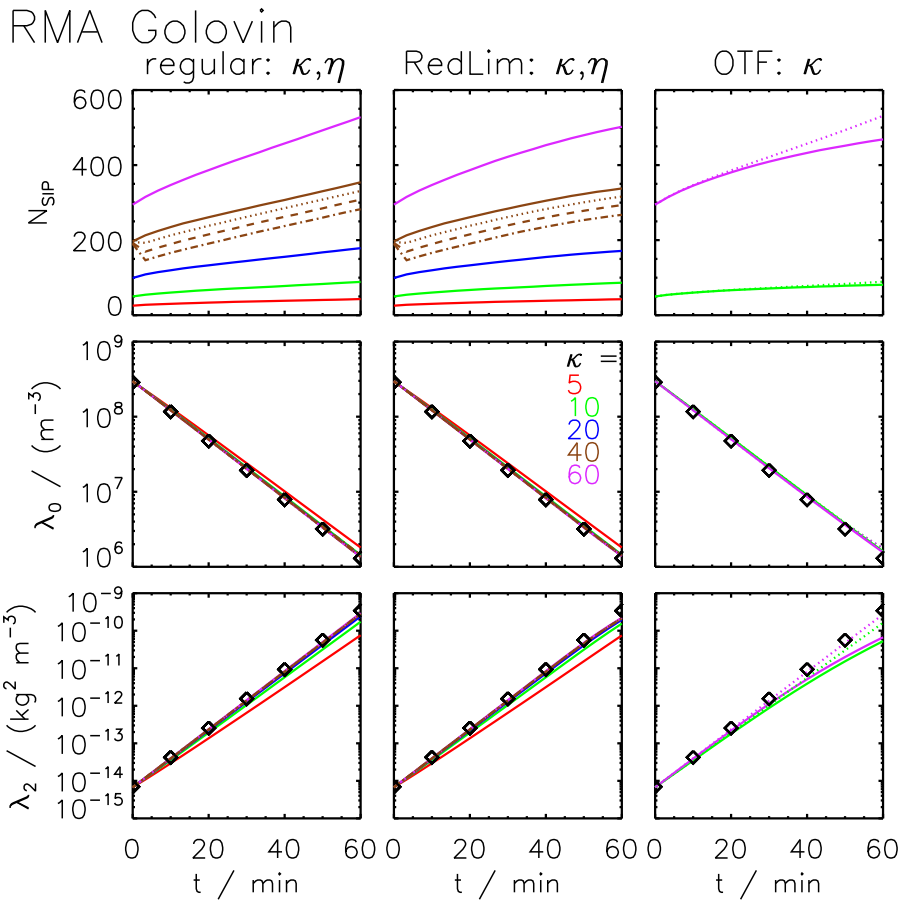

Figure 6. SIP number and moments ~~λ_0, λ_2~~ λ_0 and ~~λ_3~~ λ_2 as a function of time obtained by the RMA algorithm for the Golovin kernel. The black ~~curves~~ diamonds show the ~~moments of the~~ reference solution. ~~All other~~ The curves depict the RMA results (ensemble averages over 50 realisations). The default settings are: Probabilistic singleSIP-init with weak threshold η and $\Delta t = 1\,\mathrm{s}$. Left column: regular RMA version ~~with $\kappa = 60, 20, 10, 5$~~ for various κ-values (~~brown, blue, green, red~~see legend in the middle) and threshold ~~$\eta = 10^{-5}$~~ $\eta = 10^{-8}$, ~~10^{-6}~~, 10^{-7}, ~~10^{-8}~~ 10^{-6}, 10^{-5} (solid, dotted, dashed, dash-dotted ~~, dotted~~; shown only for $\kappa = 40$). Middle column: as in left column, but RedLim version~~with reduction limiter~~. Right column: version with update on the fly~~,~~. (solid ~~/~~lines OTF_s and dotted lines ~~: start with combinations of smallest/largest droplets,~~ OTF_l). The colours define κ as ~~before~~in the two other columns, but only $\kappa = 10$ and 60-cases are shown.

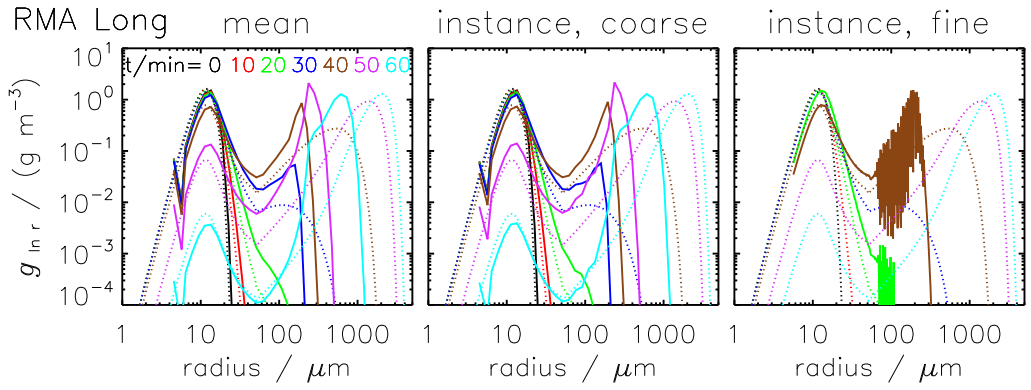

Figure 7. Mass density distributions obtained by the RMA algorithm for the Long kernel from $t = 0$ to 60 min every 10 min (from black to cyan, see legend). The ~~solid~~ dotted curves show the reference solution, the ~~dotted~~ solid curves the simulation ~~result~~ results of the RMA algorithm with Reduction Limiter ($\tilde{\gamma} = 0.1$)~~.~~ ~~The parameter settings are probabilistic singleSIP-init with~~ , weak threshold ~~,~~ $\eta = 10^{-8}$, $\Delta t = 0.1$ s and ~~$\kappa = 4$ or 20~~ $\kappa = 40$. The left panel shows the average over 50 realisations and the middle panel one specific realisation. For both, the bin resolution of the visualisation is by default $\kappa_{plot} = 4$. The right panel shows again the specific realisation (~~as indicated on top~~only $t = 20$ min and 40 min), but for $\kappa_{plot} = \kappa$.

insensitive to the bin resolution (as long as $\kappa \geq 10$), however the higher moment does not come close to the reference ~~solution (not shown). Non-oscillating results are obtained only if an unreasonably low resolution is used and very few bins exist in the problematic radius range. However, in this case, the large droplet mode does not emerge and~~ value. The effect of a Δt-variation is more substantial. Descreasing Δt, the total droplet numbers become smaller and the ~~moments are again far from the reference .~~ λ_2-values become larger, both leading to a better agreement. Despite using already a very small time step of 0.01 s in the end (we will later see that AIM and AON produce reasonable results for $\Delta t = 10$ s), the agreement with the reference solution is still not perfect.

Hence, our RMA implementation is not capable of producing reasonable results for the Long kernel.

It is not clear whether the oscillations are inherent to the original RMA algorithm or caused by the introduction of the ~~reduction limiter~~Reduction Limiter. The latter might introduce discontinuities ~~where instabilities could be triggered. The first option seems more probable, as~~ which could trigger instabilities.

At least, the Golovin RMA simulations with Reduction ~~limiter~~ Limiter do not show any ~~instability and gives a perfect agreement~~ signs of instability and agree well with the reference~~(see column 2 in Figs. 5and~~. However, this is not surprising. Clearly, the RedLim correction is only performed for SIPs, where negative weights are predicted. In Golovin simulations this happens less frequently than in Long simulations. Only in the very end, the abundance of the largest droplets is underestimated (see top right panel in Figure 5) and the increase of the higher moment levels off slightly (middle

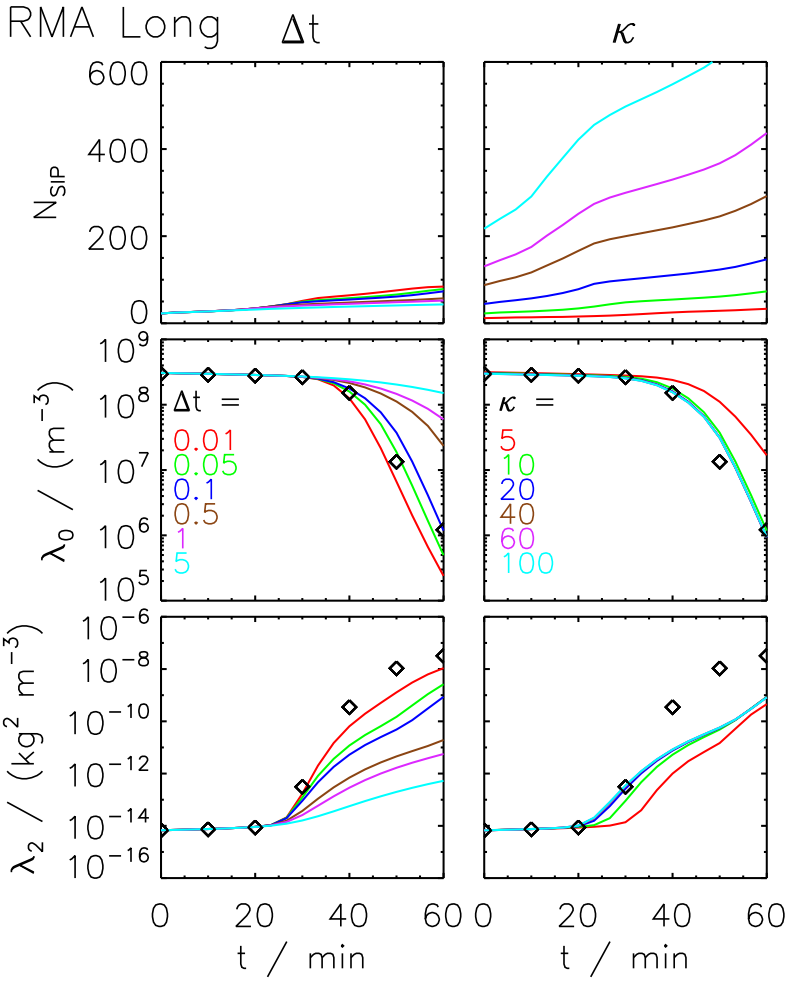

Figure 8. SIP number and moments λ_0 and λ_2 as a function of time obtained by the RMA algorithm for the Long kernel. The black diamonds show the reference solution. The curves depict the RMA results (ensemble averages over 50 realisations). The default settings are: RedLim version with $\tilde{\gamma} = 0.1$, singleSIP-init with weak threshold $\eta = 10^{-8}$, $\kappa = 10$, $\Delta t = 1\,\mathrm{s}$ and $r_{critmin} = 5.0\,\mu\mathrm{m}$. The left column shows a variation of Δt (see legend), the right one a variation of κ (see legend) .

column of Fig. 6). ~~Similarly, Golovin RMA simulations with~~ Bascially, the application of the Redlim correction, which re-scales ν_i^{Δ}, can be interpreted as an artificial reduction of the time increment (see Eq. 20) and hence slows down the growth of all corrected SIPs.

Another RMA variant uses update on the fly ~~are stable and~~ which also effectively eliminates negative weights. Such Golovin RMA simulations can be close to the reference, however the results depend on the order in which the SIP combinations are processed~~(see column 3 (and 4)in Figs. 5and~~. If collections between the smallest SIPs are treated first within each time iteration (OTF_s), then the growth of the largest droplets is too slow (see bottom left panel in Figure 5). Starting the processing with collections between the largest SIPs (OTF_l), the DSDs are as desired (see bottom right panel in Figure 5) and the moments agree perfectly with the reference if κ is sufficiently large (see right column of Fig. 6). ~~Again, Long simulations with an~~ The update on the fly ~~version of RMA are unstable (not shown)~~ has the strongest impact on those SIPs where the regular version would predict negative weights. With OTF, the weights of such SIPs strongly decrease during one time iteration and hence the continuous evaluations of the O_{ij}-values depends on the order in which the SIP combinations are processed.

Long kernel simulations with OTF_l yield results qualitatively similar to the RedLim version (see SUPP) and spurious oscillations still appear in the DSDs.

Note that the Golovin simulations used $r_{critmin} = 1.6\,\mu\mathrm{m}$, whereas the Long simulations used $r_{critmin} = 5.0\,\mu\mathrm{m}$ (note the truncated left tail in the DSDs in Figure 7). A higher $r_{critmin}$-value reduces the SIP number and the computational effort and made simulations with small time steps possible at all. The simulated λ-values are insensitive to the choice of $r_{critmin}$ (see SUPP).

We conclude that for time steps feasible in operational terms, none of the tested RMA implementations is capable of producing reasonable results with the Long kernel. Andrejczuk et al. (2010) introduced and evaluated the RMA algorithm and applied it in a simulation of boundary layer stratocumulus. Our findings are seemingly in conflict with the conclusions of their evaluation exercises. What both studies have in common is a similar trend for a κ-variation. In their Fig. 13, simulations for κ ranging roughly from 4 to 30 are depicted. The simulations with many bins show oscillations, whereas the coarsest simulation has no oscillations, but is clearly far from the real solution (largest droplets around $40\,\mu\mathrm{m}$ compared to $500\,\mu\mathrm{m}$ in the reference simulation). In their Fig. 14, they presented a detailed sensitivity test only for a $\kappa = 4$ simulation, which downplays the severity of the oscillation issue. Moreover, their simulations ran up to $2000\,\mathrm{s}$ compared to $3600\,\mathrm{s}$ in this study and many other studies (e.g. Bott, 1998; Wang et al., 2007). Hence, they missed the regime where the effect of the oscillations is strongest. Despite our extensive tests we cannot exclude that in Andrejczuk et al. (2010) an RMA implementation was used where oscillations are less cumbersome; however, the study missed to demonstrate this for a conclusive test case and we come to the conclusion that the evaluation exercises were incomplete and not suited to reveal the deficiencies faced here.

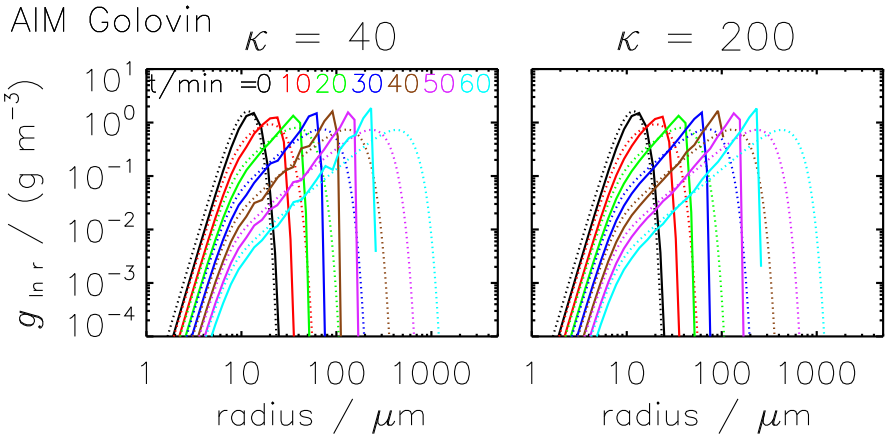

Figure 9. Mass density distributions obtained by the AIM algorithm for the Golovin kernel from $t = 0$ to 60 min every 10 min (from black to cyan, see legend). The ~~solid~~ dotted curves show the reference solution, the ~~dotted~~ solid curves the ~~simulation result of the~~ AIM ~~algorithm~~ simulation results (ensemble ~~average~~ averages over 50 realisations). The parameter settings are: probabilistic singleSIP-init with weak threshold $\eta = 10^{-9}$, ~~$\nu_{critmin} = 10^{-9} \max(\nu_i)$,~~ $\Delta t = 1$ s and $\kappa = 40$ (left) or $\kappa = 200$ (right).

RMA simulations with the Hall kernel are similarly corrupted by oscillations and do not produce useful simulations either (not shown).

3.2 Performance of Average Impact (AIM) Algorithm

Fig. 9 displays DSDs obtained by AIM for the Golovin kernel. Compared to the reference, the droplets pile up at too small radii and the algorithm is not capable of reproducing the continuous shift to larger sizes, even if a fine grid with $\kappa = 200$ (right) instead of $\kappa = 40$ (left) is used. For both κ-values, the increase of the higher moments proceeds at a too low rate (see Fig. 10), whereas the decrease in droplet number matches the analytical evolution. AIM is a very robust algorithm in the sense that the results are fairly insensitive to most numerical ~~parameters~~ parameter variations as demonstrated for κ and Δt in the left column of Fig. 10. Most simulations converge to—what we call—the best AIM solution, which is, however, not ~~the same as the~~ identical to the correct solution. The results deteriorate slightly if the initial SIP ensemble is generated with the ν_{const}-init or ν_{draw}-init instead of with the singleSIP-init (right column of Fig. 10).

The algorithm performs, in general, better for the Long and Hall kernel as is detailed in the following. Fig. 11 displays DSDs obtained by AIM for the Long kernel. Generally, the results are in good agreement with the reference solution, as long as the SIP ensemble is initialised with the singleSIP-init method (left and middle column). Towards the end of the simulated period (magenta and cyan lines), the removal of small droplets is a bit underestimated and too many small droplets are present. For $t = 30$ and 40 min, the large droplet mode is too weak as not enough large droplets have formed. At that stage, the droplets grow rapidly by collection and the AIM results lag behind. Although the

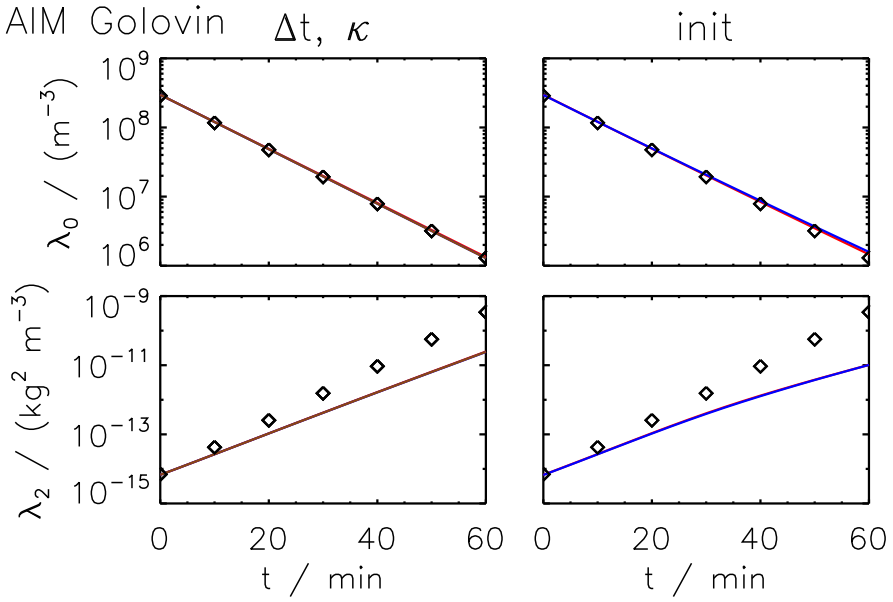

Figure 10. ~~SIP number~~ Moments λ_0 and ~~moments λ_0, λ_2 and λ_3~~ λ_2 as a function of time obtained by the AIM algorithm for the Golovin kernel. The black ~~curves~~ diamonds show the ~~moments of the~~ reference solution. ~~All other~~ The curves depict the AIM results (~~average~~ averages over 50 realisations). The default settings are: ~~Probabilistic~~ probabilistic singleSIP-init with weak threshold $\eta = 10^{-9}$, $\kappa = 40$ ~~, $\nu_{critmin} = 10^{-9} \max(\nu_i)$~~ and $\Delta t = 1\,\mathrm{s}$. Left column: default simulation (red), larger time step ($\Delta t = 10s$, blue) and more SIPs ($\kappa = 200$, brown). Right column: ν_{const}-init (red) and ν_{draw}-init (blue) with $N_{SIP} = 160$. In all panels, the curves are on top of each other.

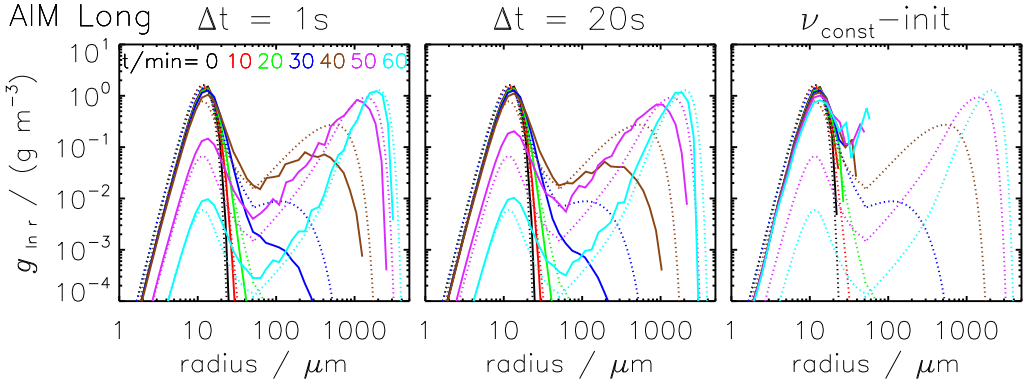

Figure 11. Mass density distributions obtained by the AIM algorithm for the Long kernel from $t = 0$ to $60\,\mathrm{min}$ every $10\,\mathrm{min}$ (from black to cyan, see legend). The ~~solid~~ dotted curves show the reference solution, the ~~dotted~~ solid curves the ~~simulation result of the~~ AIM ~~algorithm as an average~~ simulation results (ensemble averages over 50 realisations). The default settings are: ~~Probabilistic~~ probabilistic singleSIP-init with weak threshold $\eta = 10^{-9}$, $\kappa = 40$, ~~$\nu_{critmin} = 10^{-9} \max(\nu_i)$,~~ $\Delta t = 1\,\mathrm{s}$ (~~column 1~~left panel); Δt increased to $20\,\mathrm{s}$ (~~column 2~~middle panel); ν_{const}-init technique with $N_{SIP} = 160$ (~~column 3~~right panel).

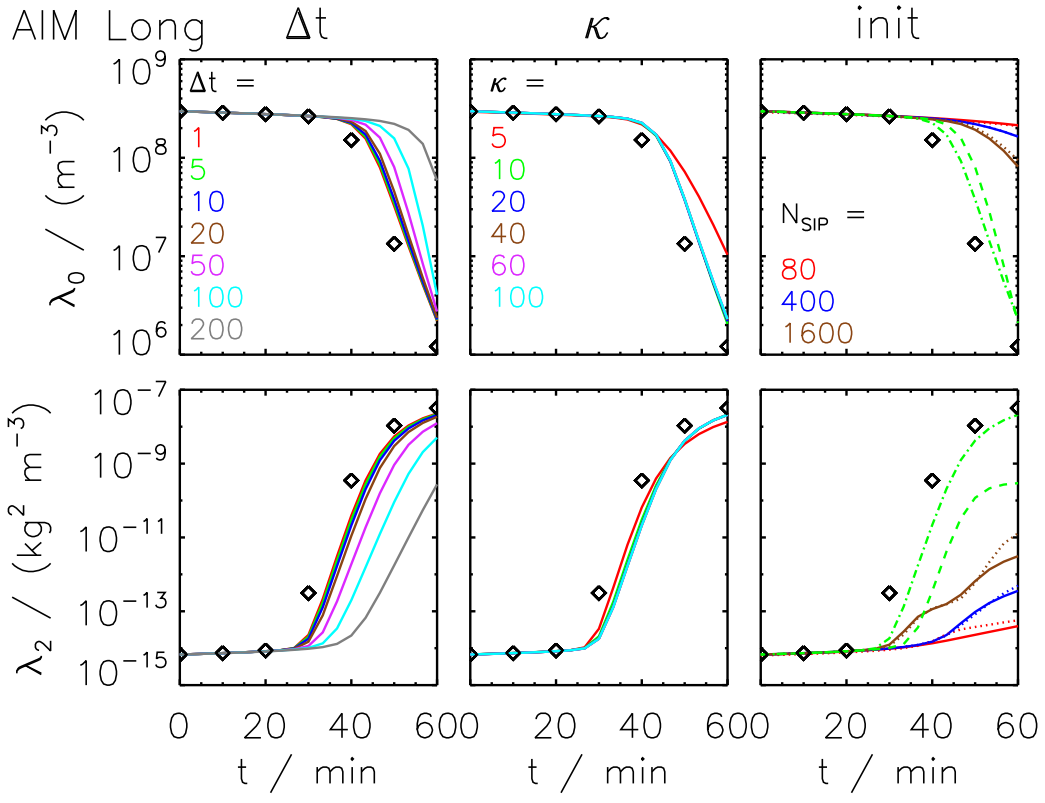

Figure 12. ~~SIP number~~ Moments λ_0 and ~~moments λ_0, λ_2 and λ_3~~ λ_2 as a function of time obtained by the AIM algorithm for the Long kernel. The black ~~curves~~ diamonds show the ~~moments of the~~ reference solution. ~~All other~~ The curves depict the AIM results (~~average~~ averages over 50 realisations). The default settings are: probabilistic singleSIP-init with weak threshold $\eta = 10^{-9}$, $\kappa = 40$ and $\Delta t = 10\,\mathrm{s}$. The left column shows a variation of ~~$\Delta t = 1, 5, 10, 20, 50, 100, 200\,\mathrm{s}$ for $\kappa = 40$. The~~ Δt (see legend) and the middle column a variation of ~~$\kappa = 5, 10, 20, 40, 60, 100$ for $\Delta t = 10\,\mathrm{s}$. Either, the default singleSIP-init~~ κ (~~solid~~see legend)~~or the singleSIP-init with $r_{critmin} = 5\,\mu\mathrm{m}$ (dotted) is used~~. The right column displays simulations with ~~different~~ various initialisation techniques~~and $\Delta t = 10\,\mathrm{s}$~~: the ν_{const}-init (solid) and ν_{draw}-init (dotted) with ~~$N_{SIP} = 1600, 400, 80$~~ various N_{SIP}-values (see legend) as well as the $\nu_{random,rs}$-init (green dashed) and $\nu_{random,lb}$-init (green dash-dotted)~~with $(\alpha_{high}, \alpha_{med}, \alpha_{low}) = (10^{-2}, 10^{-3}, 10^{-13})$ and threshold radius $r_{lb} = 16\,\mu\mathrm{m}$~~.

offset is less than five minutes, it might become crucial in simulations of short-lived clouds. Also the evolution of the moments (see Fig. 12) confirms this, as the onset of the rapid changes at around $t = 30\,\mathrm{min}$ is only slightly retarded if parameters are suitably chosen. Towards the end, the AIM results get again very close to the reference solution. The left column of Fig. 12 shows the dependence on the time step. For time steps $\Delta t \leq 20\,\mathrm{s}$ all results are similar to the best AIM solution which is close to the reference. Time steps of $50\,\mathrm{s}$ and more do not produce good enough results. Moreover, AIM is fairly insensitive to the choice of κ, $r_{critmin}$ and $\nu_{critmin}$~~(see middle column)~~. Simulations with κ ranging from 10 to 100 yield similar results (see middle column). Only, for a very coarse resolution ($\kappa = 5$) with 25 SIPs, the decrease in droplet number is too small. Increasing the lower cutoff radius $r_{critmin}$ from $0.6\,\mu\mathrm{m}$ to $5\,\mu\mathrm{m}$, the $r < 5\,\mu\mathrm{m}$-part of the DSD is represented by a single SIP and N_{SIP} is reduced by 60% (see Table 3). The predicted moments are unaffected by this variation (see SUPP). Those small-r_i SIPs are not relevant for the AIM performance. They simply carry too small fractions of the total grid box mass to be important. Their status will not change over time as already illustrated in Fig. 3. Similarly, a variation of $\nu_{critmin}$ or the switch to a strict threshold $\nu_{critmin}$ has no effect (see SUPP).

Now we draw the attention to the importance of the SIP-init method. The right panel of Fig. 11 shows the DSDs when the SIPs are initialised with the ν_{const}-init method. The algorithm completely fails and no droplets larger than $70\,\mu\mathrm{m}$ occur after 60 minutes. Consequently, the moments are far off from the reference solution (solid lines in the right column of Fig. 12). Switching to the ν_{draw}-init method (dotted lines) or using many more SIPs (up to 1600) improves the results, yet they are still useless. This clearly demonstrates how crucial the initial characteristics of the SIP ensemble are. Initialising the SIPs with an appropriate technique like the singleSIP-init, useful results are obtained with as few as 50 SIPs. Using the ν_{const}-init or ν_{draw}-init, on the other hand, solutions are still useless, even though the number of SIPs and the computation time are factor 30 and 900 higher, respectively.

The ν_{random}-simulations give another example of the importance of the init method. Even though both techniques, $\nu_{random,rs}$ (dashed line) and $\nu_{random,lb}$ (dash-dotted line), are similar in design and differ only in the creation of the largest SIPs (see Fig. 1), the outcome of the simulations is quite different. For the $\nu_{random,lb}$-init, the solution matches the best AIM solution, whereas for $\nu_{random,rs}$ the ~~moments~~ moment λ_2 ~~and λ_3 stagnate at too low levels~~stagnates at a too low level. The latter test pinpoints the main weakness of the AIM which is also reflected in its name (average impact). The initial weighting factors of those initially largest SIPs (in relation to ν of the remaining SIPs) controls how strong this growth is and how the large droplet mode emerges.

All quantities shown in Fig. 10 and 12 are averages over 50 realisations of the initial SIP ensemble. All individual realisations yield basically identical simulation results and it would have been sufficient to carry out and display simulations of a single realisation.

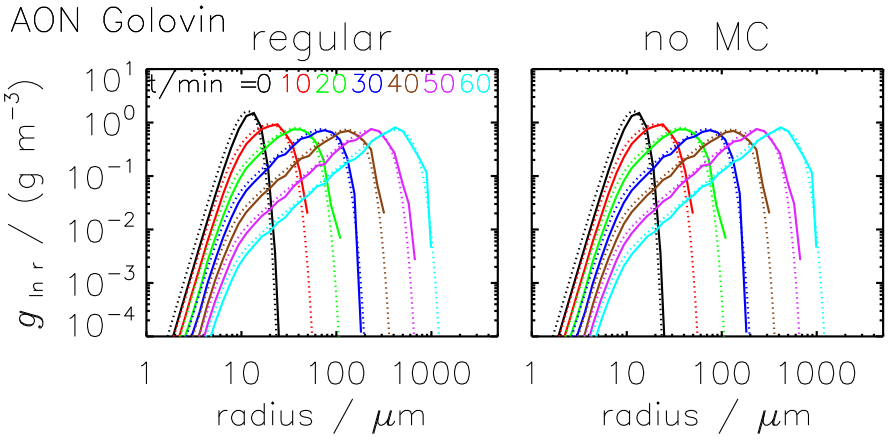

Figure 13. Mass density distributions obtained by the AON algorithm for the Golovin kernel from $t = 0$ to 60 min every 10 min (from black to cyan, see legend). The ~~solid~~ dotted curves show the reference solution, the ~~dotted~~ solid curves the ~~simulation result of the~~ AON ~~algorithm~~ simulation results (ensemble ~~average~~ averages over 50 realisations). The ~~parameter~~ default settings are: probabilistic singleSIP-init with weak threshold $\eta = 10^{-9}$, $\kappa = 40$, ~~$\nu_{critmin} = 10^{-9} \max(\nu_i)$,~~ $\Delta t = 1$ s. The ~~columns show various variants~~ left panel shows results of the regular algorithm ~~: default~~ and the right panel those of a version ~~, version~~ disregarding multiple collections~~and version disregarding self-collections (from left to right)~~.

762 ~~Figure ?? shows DSDs of~~ Next, simulations with the Hall kernel are shortly discussed (figures are
763 only shown in the supplement). Compared to the Long simulations, the reference solution reveals that
764 small droplets are much more abundant~~(see reference solution)~~, as the collection of small droplets
765 proceeds at a lower rate. This makes the simulation less challenging from a numerical point of view
766 and AIM DSDs come closer to the reference than in the Long simulations. Consequently, the AIM
767 moments agree very well with the reference~~as shown in Fig. ??.~~. For $\Delta t \leq 20$ s and $\kappa \geq 20$, all
768 solutions are similar to the best AIM solution.

769 ~~Mass density distributions obtained by the AIM algorithm for the Hall kernel from $t = 0$ to 60 min~~
770 ~~every 10 min (from black to cyan). The solid curves show the reference solution, the dotted curves the~~
771 ~~simulation result of the AIM algorithm as an average over 50 realisations. The default settings are:~~
772 ~~Probabilistic singleSIP-init, $\kappa = 40$, $\nu_{critmin} = 10^{-9} \max(\nu_i)$, $\Delta t = 1$ s (column 1); Δt increased~~
773 ~~to 20 s (column 2).~~

774 ~~SIP number and moments λ_0, λ_2 and λ_3 as a function of time obtained by the AIM algorithm for~~
775 ~~the Hall kernel. The black curves show the moments of the reference solution. All other curves depict~~
776 ~~the AIM results (average over 50 realisations). The left column shows a variation of $\Delta t = 1$, 5, 10, 20, 50, 100, 200 s~~
777 ~~for $\kappa = 40$ and the right column a variation of $\kappa = 5$, 10, 20, 40, 60, 100 for $\Delta t = 10$ s.~~

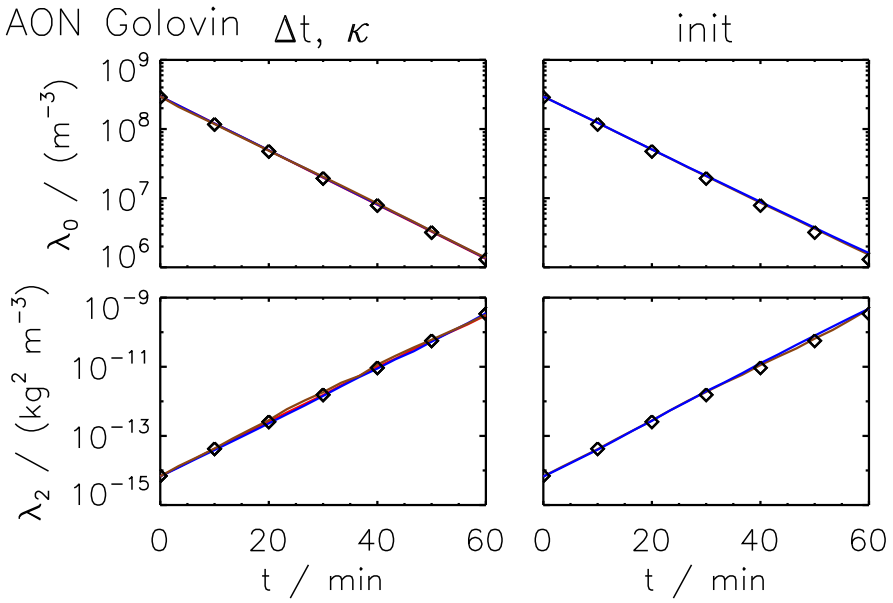

Figure 14. ~~SIP number~~ Moments λ_0 and ~~moments λ_0, λ_2 and λ_3~~ λ_2 as a function of time obtained by the AON algorithm for the Golovin kernel. The black ~~curves~~ diamonds show the ~~moments of the~~ reference solution. ~~All other~~ The curves depict the AON results (~~average~~ averages over 50 realisations). The default settings are: ~~Probabilistic~~ probabilistic singleSIP-init with weak threshold $\eta = 10^{-9}$, ~~$\kappa = 40, \nu_{critmin} = 10^{-9} \max(\nu_i)$~~ $\kappa = 40$ and ~~$\Delta t = 1$s~~$\Delta t = 1$s. Left column: default simulation (red), larger time step ($\Delta t = 20s$, blue) and fewer SIPs ($\kappa = 10$, brown). Right column: ν_{const}-init (brown) ~~,~~ and ν_{draw}-init (blue)~~and singleSIP-init with $r_{critmin} = 1.6\,\mu$m (red)~~.

3.3 Performance of All-Or-Nothing (AON) Algorithm

Fig. 13 shows the AON results for the Golovin kernel. An excellent agreement with the reference solution is found which proves at least the correct implementation of AON. Switching to a version without multiple collections (i.e. SIP i collects at most ν_i droplets in every time step) does not affect the solution as cases with $p_{crit} > 1 \Leftrightarrow \nu_k > \nu_i$ occur rarely. The AON moments closely follow the reference solution, even when the time step is increased from 1s to 10s or fewer SIPs are used ~~when~~ by decreasing κ ~~is decreased~~ from 40 to 10 (left column of Fig. 14). Unlike to AIM, AON is successful, even when the initial SIP ensemble is created with the ν_{const}-init or ν_{draw}-init (right column of Fig. 14). ~~The moments are averages over 50 realisations. For the ν_{draw}-init method, the deviation in λ_3 towards the end of the simulated period is due to a single outlier realisation where the initial values of the moments λ_2 and λ_3 were already much higher than λ_2 and λ_3 of the reference solution. Column 2 of Fig. 1 already illustrated the large uncertainty of the initial values, which becomes increasingly larger for higher order moments. Hence, this outlier behaviour is associated with a deficiency of the init technique rather than being an algorithm-intrinsic feature.~~

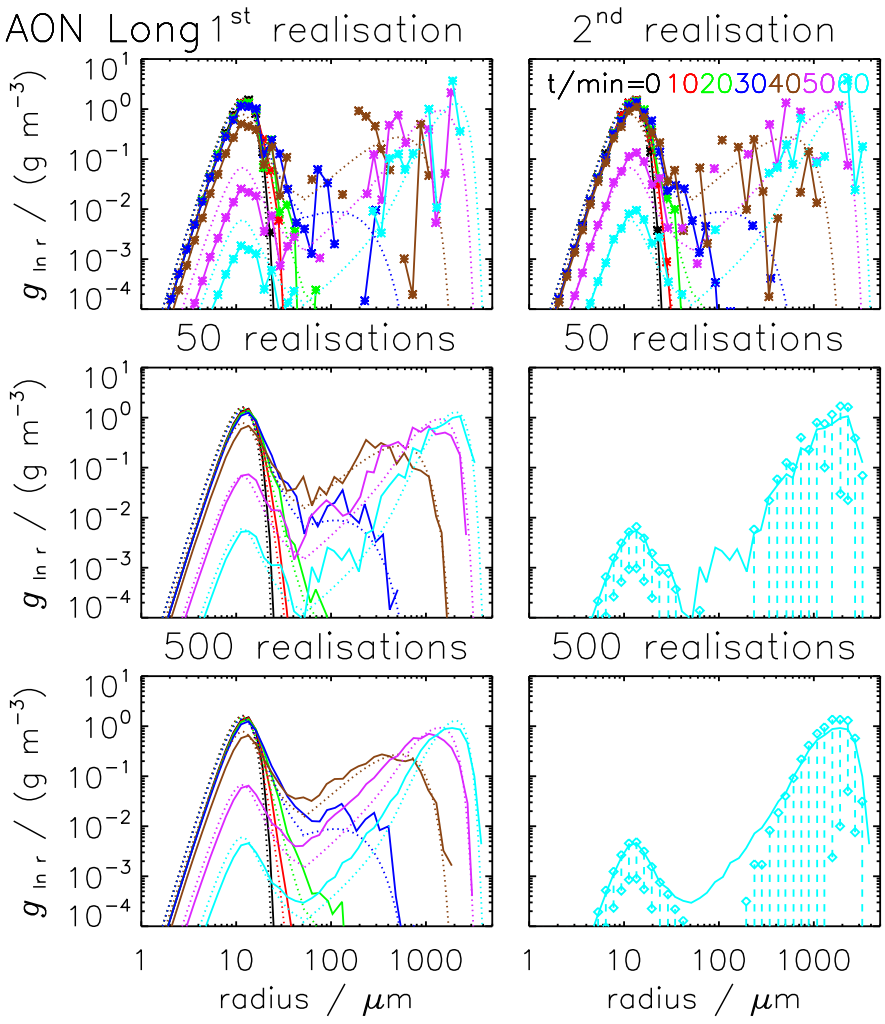

Figure 15. Mass density distributions obtained by the AON algorithm for the Long kernel from $t = 0$ to 60 min every 10 min (from black to cyan, see legend). The ~~solid~~ dotted curves show the reference solution, the ~~dotted~~ solid curves the ~~simulation result of the~~ AON ~~algorithm~~simulation results. ~~Columns 1 and 2 show individual~~ The top row shows two specific realisations (each $*$-symbol depict a non-zero g-value). ~~Columns~~Rows ~~3~~ 2 and ~~4~~ 3 show averages over 50 and 500 realisations: The left column uses the format as all DSD plots before. The right column depicts the final DSD at $t = 60$ min together For each bin, the interquartile range is determined and depicted by ~~+-symbols with~~ diamonds and a dashed bar~~(only for $t = 60$ min)~~. If there is only one ~~+-symbol~~(or none) diamond in a bin, the 25th (and the 75th) percentile is/are too small to be visible. The settings are: ~~Probabilistic~~ probabilistic singleSIP-init with $\eta = 10^{-9}$, $\kappa = 40$ ~~, $\nu_{critmin} = 10^{-9} \max(\nu_i)$,~~ and $\Delta t = 20$ s.

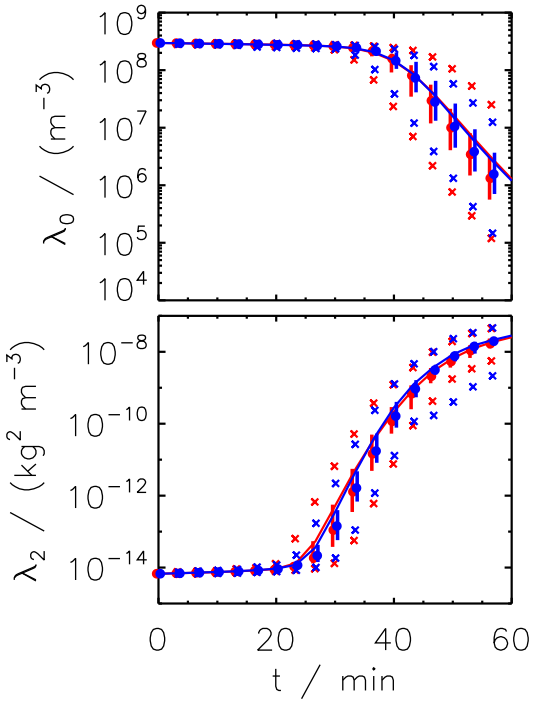

Figure 16. Moments ~~λ_0, λ_2~~ λ_0 and ~~λ_3~~ λ_2 as a function of time obtained by the AON algorithm for the Long kernel. Each realisation was initialised with a different SIP ensemble (probabilistic singleSIP, red) or all realisations started with the same SIP ensemble (deterministic singleSIP, ~~green~~blue). In both cases, the curves show an average over 50 realisations with the vertical bars indicating the interquartile range. The crosses show the minimum and maximum values and the circle the median value. The ~~black symbols depict the reference solution. The~~ parameter settings are $\Delta t = 20$ and $\kappa = 40$.

~~Nevertheless, the simulations reveal large differences between individual realisations which deserves a closer inspection.~~ Fig. 15 displays DSDs of ~~AON~~ an AON simulation for the Long kernel. The ~~two left panels~~ simulations exhibit large differences between individual realisations which deserves a closer inspection. The top row show DSDs of ~~single~~ two specific realisations. The ~~*~~*-symbol depicts the g-value for each bin. Those symbols are connected by default. An interruption of the connecting line indicates one or more empty bins ($g = 0$) where no SIPs exist in this specific radius interval. This occurs frequently ~~and the~~ due to the broadening of the DSD. The solutions are full of spikes and irregularly over- and undershoot the reference solution, particularly in the large droplet mode. The small droplet mode is underestimated in the first realisation and overestimated in the second realisation, for instance. The advantages of AON become apparent when the DSDs are averaged over many realisations as shown in ~~columns 3 and 4.~~ rows 2 and 3. Then the DSDs come close to the reference solution (left column) and the interquartile range indicates the broad envelope the individual realisations span around the reference solution (right column). Whereas the average over 50

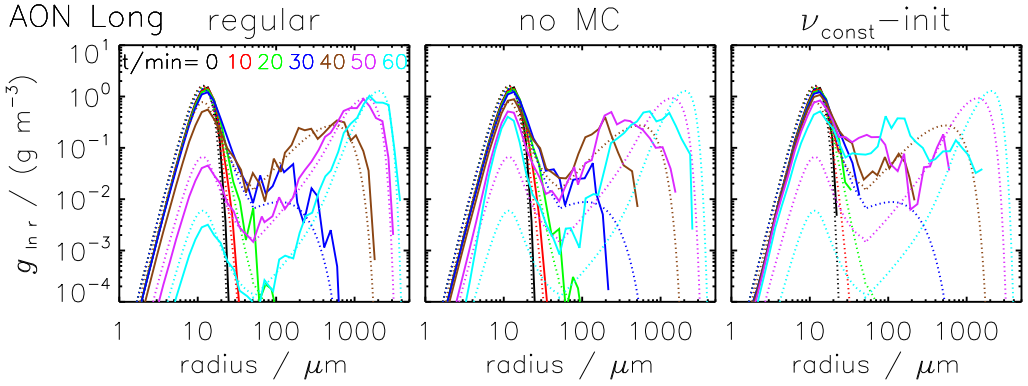

Figure 17. Mass density distributions obtained by the AON algorithm for the Long kernel from $t = 0$ to 60 min every 10 min (from black to cyan, see legend). The ~~solid~~ dotted curves show the reference solution, the ~~dotted~~ solid curves the ~~simulation result of the~~ AON ~~algorithm as an average~~ simulation results (ensemble averages over 50 realisations). The default settings are: ~~Probabilistic~~ probabilistic singleSIP-init with weak threshold $\eta = 10^{-9}$, $\kappa = 40$ ~~, $\nu_{critmin} = 10^{-9} \max(\nu_i)$,~~ and $\Delta t = 1$ s~~(column 1);~~. The left panel shows results of the regular algorithm, the middle panel those of a version disregarding multiple collections at $\Delta t = 10$ s ~~(column 2);~~ and the right panel results for ν_{const}-init ~~technique~~ with $N_{SIP} = 160$~~(column 3)~~.

realisations still has some fluctuations (row 2), the average over 500 realisations produces a smooth solution ~~.~~(row 3).

There are two sources that are potentially responsible for the large ensemble spread: the probabilistic SIP initialisation and the probabilistic AON approach. In a sensitivity test, 50 realisations are computed, all using the same SIP initialisation obtained by a deterministic singleSIPinit. Figure 16 compares those simulations to regular simulations with differing SIP initialisations. In both cases, we find a substantial ensemble spread. Starting with identical SIP initialisations the spread in terms of interquartile range is, however, somewhat smaller suggesting that both sources contribute to the ensemble spread.

Fig. 17 shows AON results with 50 realisations and probabilistic initialisation which gives a good trade-off between computational cost and representativeness. Clearly, AON DSDs are less smooth than those of AIM. Column 1 shows a default simulation with ~~singleSIP init~~ singleSIP-init and shows very good agreement with the reference solution. Disenabling multiple collections (column 2), far too few small droplets become collected and their abundance is substantially overestimated. As a consequence, the mass transfer from small to large droplets is slowed down and the large droplet mode is underestimated. Using the ν_{const}-init, the large droplet mode is not well matched and results are again useless.

Fig. 18 shows the temporal evolution of moments ~~λ_0, λ_2 and λ_3~~ λ_0 and λ_2 for a large variety of sensitivity tests. Column 1 shows a variation of Δt for the singleSIP-init. The larger Δt is chosen, the more often combinations with $p_{crit} > 1$ occur and the more crucial it becomes to consider multiple

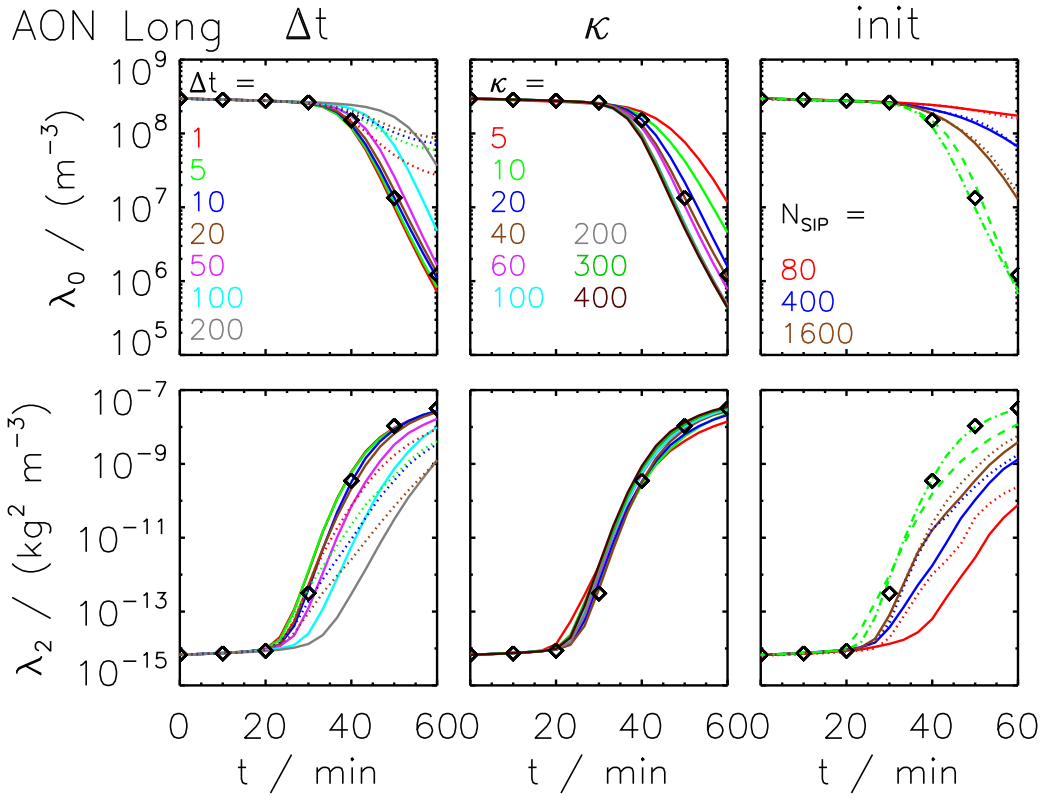

Figure 18. ~~SIP number~~ Moments λ_0 and ~~moments λ_0, λ_2 and λ_3~~ λ_2 as a function of time obtained by the AON algorithm for the Long kernel. The black ~~symbols depict~~ diamonds show the ~~moments of the~~ reference solution. ~~All coloured~~ The curves ~~show~~ depict the AON results (~~average~~ averages over 50 realisations). The default settings are: probabilistic singleSIP-init with weak threshold $\eta = 10^{-9}$, $\kappa = 40$ and $\Delta t = 10\,\mathrm{s}$. The left column shows a variation of ~~$\Delta t = 1$, 5, 10, 20, 50, 100, 200 s~~ Δt (see legend) for ~~$\kappa = 40$ for~~ the regular AON version (solid) and for a version disregarding multiple collections (dotted, only cases with $\Delta t \leq 20\,\mathrm{s}$ are displayed). The middle column shows a variation of ~~$\kappa = 5$, 10, 20, 40, 60, 100, 200, 300, 400 for singleSIP-init~~ κ (~~solid~~see legend)~~, singleSIP-init with $r_{critmin} = 1.6\,\mu$m (dashed, only for $\kappa = 60$ and 100) and MultiSIP-init (dotted, only for $20 \leq \kappa \leq 100$)~~. The right column ~~shows~~ displays simulations with various initialisation techniques: the ν_{const}-init (solid) and ν_{draw}-init (dotted) with ~~$N_{SIP} = 1600$, 400, 80. The gray~~ various N_{SIP}-values (see legend) as well as the $\nu_{random,rs}$-init (green dashed) and ~~dotted line show simulations with~~ $\nu_{random,lb}$-init ~~and $\nu_{random,rs}$-init, respectively~~(green dash-dotted).~~All simulations shown in the middle and right panel use $\Delta t = 10\,\mathrm{s}$.~~

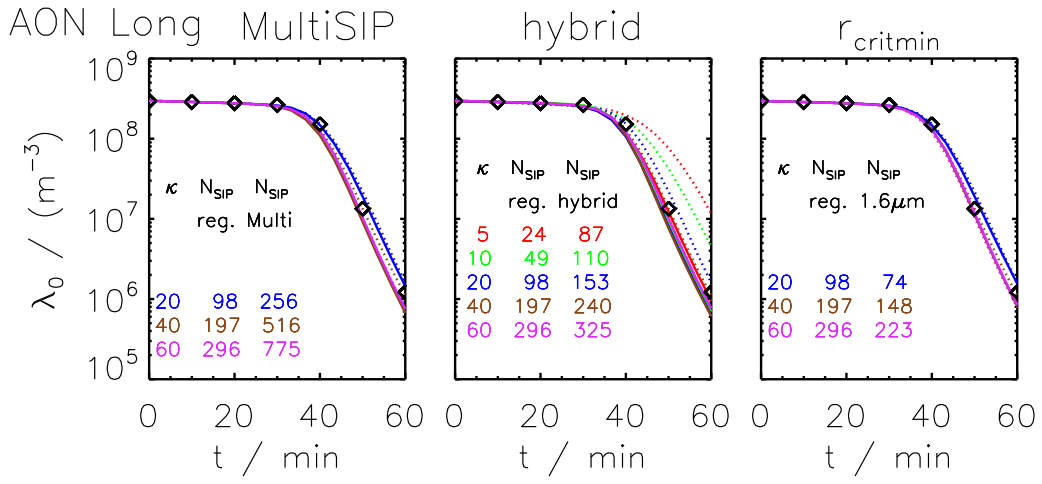

Figure 19. Droplet number as a function of time obtained by the AON algorithm for the Long kernel. The black symbols show the moments of the reference solution. In each panel, the dotted curves depict the results with the regular singleSIP-init as already shown in column 2 of Fig. 18. The solid ~~/dotted~~ curves ~~show simulations~~ depict results with a modified initialisation: the right panel shows results with the MultiSIP-init, the middle column with the hybrid ~~/regular~~ init and the right column with the singleSIP-init with $r_{critmin} = 1.6\,\mu$m. Each panel shows results for various κ-values (~~5 to 60,~~ see corresponding legend). The hybrid version uses ~~$\kappa = 100$~~ $\kappa = 100$ for radii above $15\,\mu$m and κ as labeled for radii below $15\,\mu$m. The MultiSIP-init and hybrid version ~~uses~~ use more SIPs than the regular ~~version (see~~ SingleSIP-init. An $r_{critmin}$-increase leads to a N_{SIP} ~~-values~~ -reduction. See listed N_{SIP}-values in the ~~plot)~~plots for a comparison.~~The dotted lines are identical to solid lines in col 2 of 18~~

collections. Even for the smallest time step considered, the version without multiple collections does not collect enough small droplets and hence overestimates droplet number. With the regular AON version considering multiple collections, reasonable results are obtained for time steps $\Delta t \leq 20\,$s. Column 2 shows a variation of κ for singleSIP-init. Whereas the higher moments perfectly match the reference, the droplet number shows a non-negligible dependence on κ. For $\kappa < 100$, droplet number decrease is faster, the finer the resolution is. For ~~$\kappa > 100$~~$\kappa \geq 100$, a variation of κ has no effect, hence convergence is reached. However, those simulations underestimate the droplet number. Best results are obtained for an intermediate resolution of $\kappa = 40$. Using the MultiSIP-init, the simulations show the same undesired behaviour (see left panel of Figure 19). Hence, increasing the SIP concentration in the middle part of the initial DSD has no positive effect despite using around 160% more SIPs (see N_{SIP}-values listed in the figure's legend). In another experiment, ~~the~~ a hybrid singleSIP-init was used. Below $r = 16\,\mu$m SIPs are initialised as usually with the prescribed κ. Above this radius, a high resolution with $\kappa = 100$ is always used irrespective of the chosen κ. Clearly, more SIPs are initialised with this hybrid version relative to the original version (see N_{SIP}-values listed in the figure legend). The middle panel of Figure 19 shows the droplet number evolution for the original singleSIP-init

and the new hybrid version. The sensitivity to κ is basically suppressed when the hybrid version is used. This implies that the AON algorithm is more or less insensitive to the resolution in radius range $r < 16\,\mu\mathrm{m}$, however, it is sensitive to the SIP resolution in the right tail. For example, the $\kappa = 5$-simulation with the hybrid version and 87 SIPs performs better than the $\kappa = 20$-simulation with the regular init and 98 SIPs.

In the conventional version, SIPs are initialised down to a radius of $0.6\,\mu\mathrm{m}$ (as can be seen in the top left panel of Fig. 1). Another variation of the singleSIP-init is shown in ~~column 2 of Fig. 18 (dashed curves)~~ the right panel of Figure 19 where this lower cut-off radius is raised to $1.6\,\mu\mathrm{m}$ and around $25\,\%$ fewer SIPs are used to describe the DSD. The simulation results are basically identical to the conventional init version and suggest that those initially small-r_i, small-ν_i SIPs are not relevant for the performance of AON.

Further tests with the singleSIP-init include a variation of the threshold parameter η and a switch from weak thresholds to strict thresholds. Moreover, we investigated the implications of update-on-the-fly of the SIP properties. The singleSIP-init produces an initially radius-sorted SIP ensemble and looping over the i-j combinations in the algorithm starts with combinations of the smallest droplets, which may introduce a bias. We reversed the order (i.e. started with largest droplet combinations) or randomly rearranged the order of the SIP combinations. None of those variations had a significant effect on the ~~results (not shown).~~ ensemble-averaged results (see SUPP). The latter insensitivity is in contrast to the RMA behaviour. The reason for this is the comparably small number of SIP combinations that actually result in collections, as well as probabilistic determination of these combinations. This prevents any pronounced bias due to size-sorting. Moreover, AON does not preserve the size-sortedness of the SIP list (cf. Fig. 4).

Finally, the AON performance for other SIP initialisations is discussed (right column of Fig. 18). As already demonstrated in Fig. 17, AON is not able to produce a realistic large droplet mode, if a moderate number of SIPs is initialised with the ν_{const}-technique. Hence, the higher moments are underestimated and droplet number is overestimated. Increasing the number of SIPs up to 1600, the solutions get closer to the reference, yet the agreement is still not satisfactory. The performance for the ν_{draw}-init is similar. Keeping in mind the previous sensitivity studies (hybrid singleSIP-init, MultiSIP-init), it is apparent that the ν_{const}-init and ν_{draw}-init suffer from an undersampling of the initially largest droplets. Due to its simplicity, using constant weights for initialisation has been a common approach in previous 3D-LCM cloud simulations (Shima et al., 2009; Hoffmann et al., 2015). Hence, we tested AON extensions aiming at a better performance for such equal weights initialisations.

Let us consider the possible weighting factors the SIPs can attain in the course of a simulation. In the beginning, all SIPs have $\nu = \nu_{init}$. After a collection event, for both involved SIPs $\nu = \nu_{init}/2$. If such a $\nu = \nu_{init}/2$-SIP collects a $\nu = \nu_{init}$-SIP, both SIPs carry $\nu_{init}/2$ droplets. Subsequent collections can generate SIPs with weighting factors $\nu_{init}/4$, $3\,\nu_{init}/4$ and so on. It may be advantageous,

if AON generates a broader spectrum of possible ν-values and produces SIPs with smaller weights more efficiently. So far, the equal splitting approach with $\xi = 0.5$ in a collection event of two equal-ν SIPs has been used. In sensitivity tests, a random number for ξ is drawn in each collection event, either from a uniform distribution $\xi \in [0,1]$ or from a log-uniform distribution $\xi \in [10^{-10}, 10^{0}]$. Enhancing the spread of ν-values, more collection events occur in the algorithm, as p_{crit} is ~~smaller~~ larger when small-ν SIPs are involved. Once most SIPs were part of a collection event, the first option with $\xi \in [0,1]$ produces a distribution of ν-values that is similar to the initial ν-distribution of the ν_{draw}-init technique and further equal weights combinations are unlikely to occur. Hence, the new version does not improve the simulation results, as the outcome for the ν_{draw}-init and the standard ν_{const}-init are similar (~~not shown~~see SUPP). Other variations produce smaller weights with $\xi = 10^{-10\ rand()}$ or $\xi = 10^{-10\ rand()^2}$, yet without any noticeable improvement in the simulation results (~~not shown~~see SUPP).

To complete the analysis for the Long kernel, the right column of Fig. 18 shows simulation results for $\nu_{random,lb}$ and $\nu_{random,rs}$. In short, AON can cope with those initialisations and produces useful results.

As already noted in the AIM section, Hall simulations are not as challenging as Long simulations from a numerical point of view. As the collection of small droplets proceeds at a lower rate for the Hall kernel, disenabling multiple collections in the AON simulations does not deteriorate the results as much as in the Long simulations (~~not shown~~see SUPP). Besides this, simulations with the Hall kernel lead to similar conclusions as for the Long simulations and are therefore not discussed in more detail.

4 Discussion

The presented box model simulations can be regarded as a first evaluation step of collection/aggregation algorithms in LCMs. The final goal is the evaluation in (multi-dimensional) applications of LCMs with full microphysics. In order to isolate the effect of collection, other microphysical processes like droplet formation and diffusional droplet growth have been switched off and all box model simulations started with a prescribed SIP ensemble following a specific exponential distribution. In section 4.1 the performance of the different algorithms is compared and we summarise the findings from section 3. Section 4.2 discusses implications of our results and provides further insights.

4.1 Summarising comparison of the algorithms' performance

The initialisation techniques for the SIP population generation are mostly probabilistic and by default, each simulation was performed for 50 different realisations. For RMA and AIM, we found the ensemble spread to be small and a single realisation is as good as the ensemble mean. The AON algorithm is inherently probabilistic and we highlighted the substantial ensemble spread. Reasonable

results are only obtained only by averaging over many realisations. One may argue that this precludes the usage of AON in real-world applications as it is not feasible to run 50 realisations in each grid box of a 2D/3D model simulation. However, we are not that pessimistic. In such simulations, many grid boxes have similar atmospheric conditions and averaging will occur across such grid boxes. We made a similar experience in simulations of contrail-cirrus, where we tested the N_{SIP}-sensitivity of the deposition/sublimation process (see section 3.1 in Unterstrasser and Sölch, 2014) . We found that very few SIPs per grid box sufficed to reach convergence even though the few SIPs in a single grid box could not realistically represent a smooth DSD and reasonable DSDs could only be obtained by averaging over several grid boxes.

RMA simulations for the Long kernel require around a factor 1000 smaller time steps than the respective AON and AIM simulations ($\Delta t = 0.01\,\mathrm{s}$ versus $10\,\mathrm{s}$). Using the Long kernel, rapid collection growth occurs in a certain size range. In RMA, this puts a strong constraint on the time step (see Eq. 24). In AON the inclusion of multiple collections allows simulating the rapid growth without the need to reduce the time step. Without multiple collections, the AON requirements on Δt would be similar to RMA. AIM seems to be unaffected by rapid collections resulting in negative weighting factors as observed in RMA. The reason for this might origin from AIM's typical behavior. If large and therefore most effectively collecting SIPs are produced at all, they will exhibit very small weighting factors. This property reduces the potentially hazardous impact of multiple collections at larger time steps in the tested setups. However, this might not be a universal feature of AIM.

If the initial SIP ensemble is created with the SingleSIP-init, 50 to 100 SIPs are needed for convergence in any of the three algorithms. This value is similar to the number of bins used in traditional algorithms for spectral-bin models (Bott, 1998; Wang et al., 2007) .

For a given N_{SIP}, the number of floating point operations performed in one time iteration is roughly similar for all three algorithms but depends ultimately on details of the implementations. The RMA RedLim variant is, e.g., more demanding than its OTF counterpart. In the AON algorithm, the generation of the random numbers needs a non-negligible share of the computing time.

The time complexity of all presented algorithms is $\mathcal{O}(N_{SIP}^2)$ as computations are carried out for all pairwise combinations of SIPs. A linear sampling approach as introduced by Shima et al. (2009) , which processes only $N_{SIP}/2$ SIP pairs, has complexity $\mathcal{O}(N_{SIP})$ and can be applied in the RMA or AON algorithm. However, more SIPs may be required to reach convergence and in full microphysical models this may slow down the calculation of all other microphysical processes (which have usually linear time complexity).

All in all, the time step Δt, which controls the number of iterations, is the most critical parameter for the computing time.

4.2 Implications and further insights

In this section, we provide further insight and discuss the implications from the box model tests. Since our results have been gained with typical assumptions for warm clouds, we discuss their representativeness for ice clouds.

The evaluation of different initialisation methods showed that the performance of the collection/aggregation approaches depends essentially on the way the SIPs are initialised, a problem which is inherently absent in spectral-bin models. Their initialisation resembles the singleSIP technique used here, i.e. the number concentration (the weighting factor) within a bin (for a certain mass range represented by one SIP) is directly prescribed. However, LCMs exhibit a larger variety of how an initial droplet spectrum can be translated into the SIP space. The study showed that the singleSIP is advantageous for the correct representation of the collisional growth, since they initialise large SIPs with small weighting factors, which are responsible for the strongest radius growth. On the other hand, the ν_{const} initialisation technique, in which all SIPs have the same weighting factor initially as it is done in many current (multi-dimensional) applications of LCMs, impedes significantly the correct representation of collisional growth.

In this idealised study, we were able to control (to a certain extent) the representation of droplet spectra by various initialisation methods. In more-dimensional simulations with full microphysics, however, this is not straightforward nor has it been intended. So far, convergence tests in "real-world" LCM applications simply included variations of the SIP number and have not focused on more detailed characteristics of the SIP ensemble (i.e. the properties that have been discussed in Fig. 1). Droplet formation and diffusional droplet growth, which usually create the spectrum from which collisions are triggered, should be implemented such that "good" SIP ensembles are generated or evolve before collection becomes important. Here, good refers to a SIP ensemble for which the collection/aggregation algorithm performs well. For instance, the basic idea of the ~~initialisation technique~~ ν_{random}~~, the initialisation of weighting factors~~ -initialisation technique (weighting factors are uniformly distributed in $\log(\nu)$~~,~~) might also improve multi-dimensional simulations.

Generally, the performance of the algorithms is better when the SIP ensemble features a broad range of weighting factors. One viable option to achieve this is the introduction of a SIP splitting technique (Unterstrasser and Sölch, 2014). ~~Why~~ How this may improve the performance of the collection/aggregation algorithms is outlined next.

Mass fractions represented by individual SIPs, $\tilde{\chi}_i$, are analysed. $\tilde{\chi}_i$ is defined as $\chi_i/\mathcal{M}$, i.e. the total droplet mass in a SIP χ_i is normalised by the total mass within the grid box $\mathcal{M}$. Figure 20 shows the initial $\tilde{\chi}_i$-values ~~for~~ of all SIPs as a function of their initial radius r_i. Results are shown for AIM and AON with the singleSIP-init method and two bin resolutions $\kappa = 20$ and 100~~as a function of their initial radius r_i.~~. This corresponds to 99 and 493 SIPs for the specific realisation depicted here. The two rows show the same data, using a logarithmic (top row) or linear y-scale (bottom). The log scale version highlights that $\tilde{\chi}_i$-values spread over many orders of magnitudes. Mainly, the parameter $\nu_{critmin}$ controls the minimum value of χ_i. The heaviest SIPs carry initially up to 6.5% ($\kappa = 20$) or

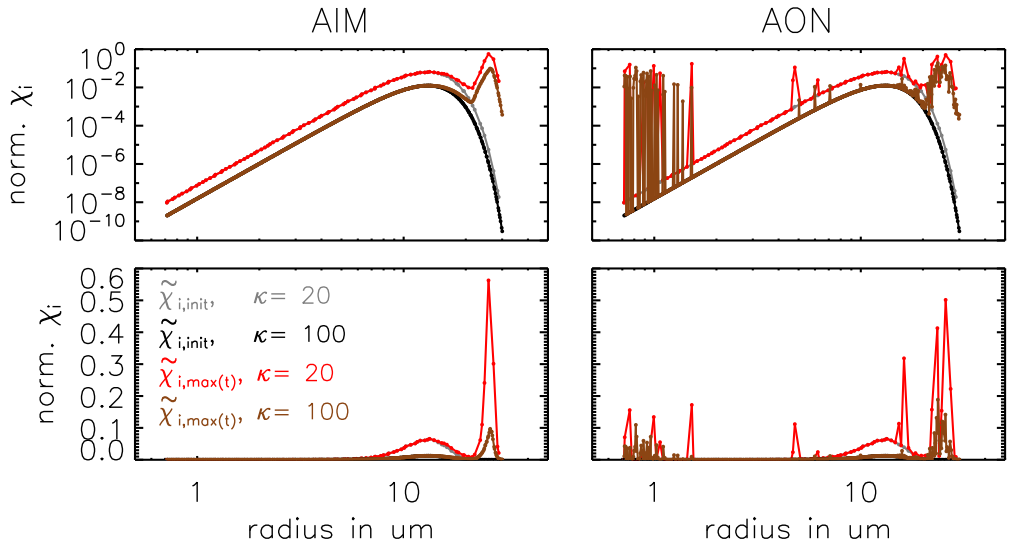

Figure 20. Normalised SIP mass $\tilde{\chi}_i$ as a function of the initial SIP radius r_i. $\tilde{\chi}_i$ is defined as $= \chi_i / \mathcal{M} = (\nu_i \mu_i)/\mathcal{M}$, i.e. the total droplet mass in a SIP is normalised by the total mass within the grid box. ~~χ_{init}~~ $\tilde{\chi}_{init}$ denotes $\tilde{\chi}_i$ of the initial SIP ensemble. ~~χ_{max}~~ $\tilde{\chi}_{max}$ denotes the maximum $\tilde{\chi}_i$-value each SIP attains over the course of a simulation. The left/right panel shows AIM/AON simulations with $\kappa = 20$ or 100 (see legend). Both algorithms use the singleSIP-init ~~,~~ and $\Delta t = 10\,\mathrm{s}$. The plots show results from a single realisation.

1.2% ($\kappa = 100$) of the total mass $\mathcal{M}$ (see bottom row). Clearly, the values of the $\kappa = 20$-simulation are larger, as the total mass is distributed over fewer SIPs. For each SIP, $\tilde{\chi}_i$ is tracked over time and the maximum value, $\tilde{\chi}_{i,max(t)}$, is recorded (red and brown curves in the graphs). Characteristically of AIM, only the largest SIPs grow substantially and collect mass from other SIPs. Hence, only χ_i of those SIPs increases. By the way, this also illustrates that the χ_i-values of the smallest SIPs are so small that all those SIPs can be merged into a single SIP without changing the AIM outcome (see $r_{critmin}$-variation before). Using the fine resolution ($\kappa = 100$), heavy SIPs (i.e. those with largest $\tilde{\chi}_i$) carry up to 10% of the total grid box mass at some point in time. In the $\kappa = 20$-simulation, this ratio can be higher than 50%, meaning that one specific SIP accumulated more than 50% of the total grid box mass at some time. Hence, the grid box mass is distributed fairly unevenly over the SIP ensemble. Astonishingly, this has no effect on the performance of AIM as the predicted $\lambda_{k,SIP}$-values for both AIM simulations are basically identical (see middle column of Fig. 12). In the AON simulations, we similarly find that the grid box mass is unevenly distributed over the SIP ensemble. Different to AIM, also many initially small SIPs and a few initially medium-sized SIPs carry a relevant portion of the grid box mass at some time. The algorithms may converge better if those heavy SIPs are split into several SIPs during the simulation.

In all simulations so far, the mean radius of the initial DSD was $9.3\,\mu$m~~where~~. Then the abundance of droplets larger than around $10\,\mu$m drops strongly, which poses a challenge to ~~the representation~~

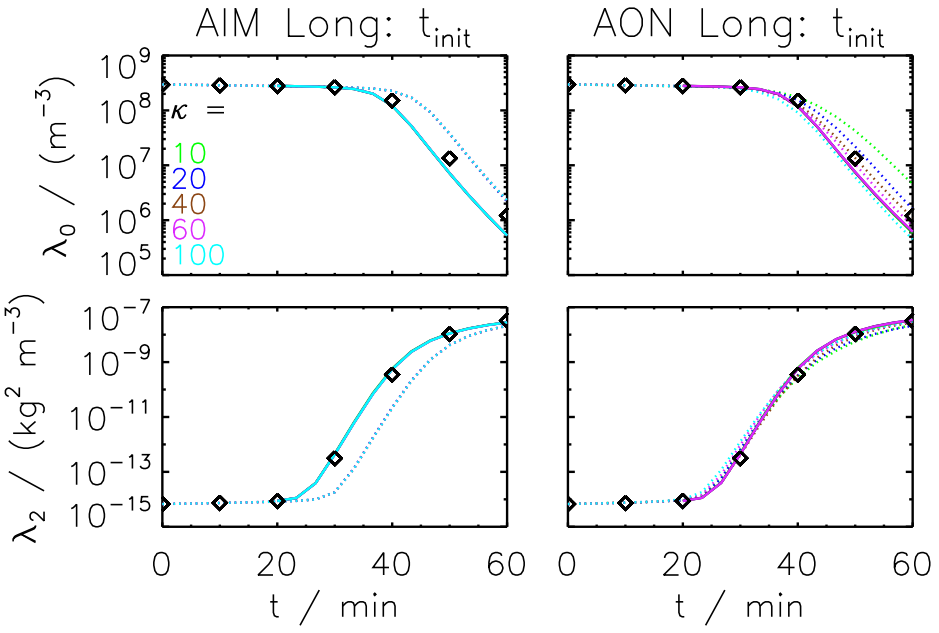

Figure 21. ~~SIP number~~ Moments λ_0 and ~~moments λ_0, λ_2 and λ_3~~ λ_2 as a function of time obtained for the Long kernel by AIM (left) and AON (right). The black symbols depict the moments of the reference solution. The simulations are initialised with Wang's solution after ~~10 (dotted),~~ 20 minutes (~~dashed~~solid lines) ~~or 30 (dash-dotted) minutes~~ using the singleSIP-init with various κ-values (see legend). The default AON and AIM simulations initialised at $t = 0$, which have been shown before in Figs. 12 and 18, are depicted by ~~solid~~ dotted lines.

representing this part of the droplet spectrum in SIP space. In a sensitivity test, we start with ~~"more~~ more "mature" DSDs. The simulations are initialised with ~~Wang's reference solution~~ the reference solution from Wang et al. (2007) after $t_{init} = 10$, 20 or 30 minutes (cf. red, green and blue solid curves in previous plots of mass density distributions) using the singleSIP-init. Fig. 21 shows ~~the SIP number and various moments~~ λ_0 and λ_2 of the DSD for AIM and AON for $t_{init} = 20\,\mathrm{min}$ and the default $t_{init} = 0\,\mathrm{min}$ (cases $t_{init} = 10$ and $30\,\mathrm{min}$ are shown in SUPP). The initial DSD is broader for a later initialisation time and hence more SIPs are initialised for a given κ (see Table 3 for the resulting N_{SIP}-values). This implies in particular that the spectrum above $10-20\,\mu\mathrm{m}$ is sampled with more SIPs. For both algorithms, the simulation results are close to the reference solution. Compared to the default $t_{init} = 0$-case, a much weaker κ-dependence of the AON predicted droplet number is apparent and the AIM results do not lag behind. Even though this sensitivity test cannot be repeated for other init methods (as they require an analytical description of the initial DSD), the ~~single~~SIP singleSIP-init simulations already indicate that the SIP initialisation is not as crucial when a later initialisation time is chosen and that our default setup with a narrow DSD may overrate the importance of the SIP initialisation. What are the implications of this for simulations with full

microphysics? Clearly, the $t_{init} = 20\,\mathrm{min}$ and $30\,\mathrm{min}$-case oversimplify the problem, as such DSDs cannot be produced by diffusional growth only. The $t_{init} = 10\,\mathrm{min}$-DSD, on the other hand, is still close to the $t_{init} = 0\,\mathrm{min}$-DSD and may be produced by diffusional growth. RMA simulations with non-zero t_{init} again show spurious oscillations and fail to predict the higher moments correctly (see SUPP).

In multi-dimensional models, collection/aggregation might be further influenced by the movement of SIPs due to sedimentation or flow dynamics. For instance, sedimentation removes the largest SIPs with the potentially smallest weighting factors, while turbulent mixing ~~is able to~~ may add SIPs with their initial weighting factor into matured grid boxes, where collection has already decreased the weighting factors of the older SIPs. Indeed, the additional variability in more-dimensional simulations might compensate for the missing variability in the weighting factors usually present in simulations using the ν_{const} ~~initialisation~~ initialisation technique.

It is not clear which findings of our evaluation efforts are the most relevant aspects that control the performance of collection/aggregation algorithms in more complex LCM simulations. Nevertheless, the idealised box simulations are an essential prerequisite towards more comprehensive evaluations as they disclosed the potential importance of the SIP initialisation (an aspect that is inherently absent in spectral bin models). All in all, we can state that the behaviour of Lagrangian collection algorithms in more complex simulations demands further investigation. Nevertheless, we have already learned a lot from the box model simulations. A summary will be given in the concluding section.

Besides the academic Golovin kernel, our simulations used the hydrodynamic kernel with collection efficiencies that are usually employed for ~~liquid~~ warm clouds (Long and Hall). We found that Hall simulations are not as challenging as Long simulations from a numerical point of view. For ice clouds, usually a constant aggregation efficiency E_a (the analogon to collection efficiency E_c) is chosen, partly due to the lack of better estimates (Connolly et al., 2012). AON simulations with $E_a = 0.2$ indicated that using a constant efficiency makes the computational problem less challenging, e.g. we find a smaller sensitivity to κ compared to the Long simulations shown in Fig. 18 (~~not shown~~see SUPP). Hence, the presented algorithms can be equally employed for aggregation. Certainly, the assumption of spherical particles used here is overly simplistic for ice cloud, in particular, if aggregates form. However, including mass-area relationships (e.g. Mitchell, 1996; Schmitt and Heymsfield, 2010) in the kernel expression and using parameterisations of ice crystal fall speed (e.g. Heymsfield and Westbrook, 2010) should not change the nature of the problem.

5 Conclusions

In the recent past, Lagrangian cloud models (LCMs), which use a large number of simulation particles (SIPs, also called super droplets in the literature) to represent a cloud, have been developed and become more and more popular. Each SIP represents a certain number of real droplets~~, which~~ ; this

number is termed the weighting factor (or multiplicity) of a SIP. In particular, the collision process leading to coalescence of cloud droplets or aggregation of ice crystals is implemented differently in the various models described in the literature. The present study evaluates the performance of three different collection algorithms in a box model framework. All microphysical processes except collection/aggregation are neglected and an exponential droplet mass distribution is used for initialisation. The box model simulation results are compared to analytical solutions (in the case of the Golovin kernel) and to a reference solution obtained from a spectral bin model approach by Wang et al. (2007) (in the case of the Long or Hall kernel).

LCMs exhibit a large variety of how an initial droplet spectrum can be translated into the SIP space and various initialisation methods are thoroughly explained. The performance of the algorithms depends crucially on details of the SIP initialisation and various characteristics of the initialised SIP ensemble (an issue that is inherently absent in spectral bin models and has not been paid much attention in previous LCM studies).

The Remapping Algorithm (based on ideas of Andrejczuk et al., 2010) ~~showed~~ produces perfect solutions in simulations with the Golovin kernel, however shows a poor performance ~~, either no~~ when we switch to the Long kernel. Spurious oscillations occur in the intermediate radius range $[100\,\mu\mathrm{m}, 200\,\mu\mathrm{m}]$ which impedes the development of a realistic rain mode~~developed or the solutions became unstable~~. Only for unfeasibly small time steps of $0.01\,\mathrm{s}$, the simulation results get close to the reference solution. The evaluation exercises presented in Andrejczuk et al. (2010) were not suited to reveal ~~the obvious~~ these shortcomings or downplayed its severity. Based on our extensive tests, ~~the algorithm cannot be recommended~~ we cannot recommend the algorithm at its present state for further LCM applications, unless ~~the stability issue is solved~~some mechanism to eliminate those oscillations is developped.

The Average Impact (AIM) algorithm (based on ideas of Riechelmann et al., 2012) can produce very good results, however, appears to be inflexible inasmuch as only the initially largest SIPs are allowed to grow in radius space. The performance depends on details of the SIP initialisation much more than, e.g. on the time step or the SIP number.

The probabilistic All-or-Nothing (AON) algorithm (based on ideas of Shima et al., 2009; Sölch and Kärcher, 2010) yields the best results and is the only algorithm that can cope with all tested kernels. Unlike to AIM, in AON it is not pre-determined which SIPs will eventually contribute to the large droplet mode. By design, any SIP can become significant at some point and the algorithm can cope with SIP initialisations that guarantee a broad spectrum of weighting factors. If an equal weights initialisation is used, tremendously many SIPs are necessary for AON convergence as reported by ~~(Shima et al., 2009) .~~ Shima et al. (2009) .

Many current (multi-dimensional) applications of LCMs use such SIP ensembles with a narrow spectrum of weighting factors causing a poor performance of the collection/aggregation algorithms. This should be clearly avoided in order to have collection/aggregation algorithms to work properly

and/or efficiently. The time step and the bin resolution κ (used in the singleSIP-init) have values similar to those used in traditional spectral-bin models and hence the computational efforts of both approaches for the collection/aggregation treatment are in the same range. The presented box model simulations are a first step towards a rigourous evaluation of collection/aggregation algorithms in more complex LCM applications (multidimensional domain, full microphysics).

6 Code availability

The programming language IDL was used to perform the simulations and produce the plots. The source code can be obtained from the first author. Pseudo-code of the algorithms is given in the text.

7 Competing interests

The authors declare that they have no conflict of interest.

Acknowledgements. The DFG (German Science Foundation) partly funded the first author (contract number UN286/1-2) and the second author (RA617/27-1). We thank A. Bott for providing us with his fortran code, L.-P. Wang for simulation data, M. Andrejczuk, S. Shima, I. Sölch and P. L'Ecuyer for discussions.

References

Albrecht, B. A.: Aerosols, cloud microphysics, and fractional cloudiness, Science, 245, 1227–1230, 1989.

Andrejczuk, M., Reisner, J. M., Henson, B., Dubey, M. K., and Jeffery, C. A.: The potential impacts of pollution on a nondrizzling stratus deck: Does aerosol number matter more than type?, J. Geophys. Res., 113, D19 204, doi:0.1029/2007JD009445, 2008.

Andrejczuk, M., Grabowski, W. W., Reisner, J., and Gadian, A.: Cloud-aerosol interactions for boundary layer stratocumulus in the Lagrangian cloud model, J. Geophys. Res., 115, D22 214, doi:10.1029/2010JD014248, 2010.

Andrejczuk, M., Gadian, A., and Blyth, A.: Stratocumulus over SouthEast Pacific: Idealized 2D simulations with the Lagrangian Cloud Model, ArXiv e-prints, 1211.0193v1 [physics.ao-ph], 2012.

Arabas, S. and Shima, S.-i.: Large-Eddy Simulations of Trade Wind Cumuli Using Particle-Based Microphysics with Monte Carlo Coalescence, J. Atmos. Sci., 70, 2768–2777, doi:10.1175/JAS-D-12-0295.1, http://dx.doi.org/10.1175/JAS-D-12-0295.1, 2013.

Arabas, S., Jaruga, A., Pawlowska, H., and Grabowski, W. W.: libcloudph++ 1.0: a single-moment bulk, double-moment bulk, and particle-based warm-rain microphysics library in C++, Geosci. Model Dev., 8, 1677–1707, doi:10.5194/gmd-8-1677-2015, http://www.geosci-model-dev.net/8/1677/2015/, 2015.

Ayala, O., Rosa, B., and Wang, L.-P.: Effects of turbulence on the geometric collision rate of sedimenting droplets. Part 2. Theory and parameterization, New Journal of Physics, 10, doi:10.1088/1367-2630/10/7/075016, 2008.

Bayewitz, M. H., Yerushalmi, J., Katz, S., and Shinnar, R.: The Extent of Correlations in a Stochastic Coalescence Process, J. Atmos. Sci., 31, 1604–1614, doi:10.1175/1520-0469(1974)031<1604:TEOCIA>2.0.CO;2, 1974.

Berry, E. X.: Cloud Droplet Growth by Collection, J. Atmos. Sci., 24, 688–701, http://dx.doi.org/10.1175/1520-0469(1967)024<0688:CDGBC>2.0.CO;2, 1967.

Berry, E. X. and Reinhardt, R. L.: An Analysis of Cloud Drop Growth by Collection Part II. Single Initial Distributions, J. Atmos. Sci., 31, 1825–1831, doi:10.1175/1520-0469(1974)031<1825:AAOCDG>2.0.CO;2, http://dx.doi.org/10.1175/1520-0469(1974)031<1825:AAOCDG>2.0.CO;2, 1974.

Bott, A.: A Flux Method for the Numerical Solution of the Stochastic Collection Equation, J. Atmos. Sci., 55, 2284–2293, doi:10.1175/1520-0469(1998)055<2284:AFMFTN>2.0.CO;2, http://dx.doi.org/10.1175/1520-0469(1998)055<2284:AFMFTN>2.0.CO;2, 1998.

Chen, S., Bartello, P., Yau, M. K., Vaillancourt, P. A., and Zwijsen, K.: Cloud Droplet Collisions in Turbulent Environment: Collision Statistics and Parameterization, J. Atmos. Sci., 73, 621–636, doi:10.1175/JAS-D-15-0203.1, 2016.

Connolly, P. J., Emersic, C., and Field, P. R.: A laboratory investigation into the aggregation efficiency of small ice crystals, Atmos. Chem. Phys., 12, 2055–2076, doi:10.5194/acp-12-2055-2012, http://www.atmos-chem-phys.net/12/2055/2012/, 2012.

DeVille, R., Riemer, N., and West, M.: Weighted Flow Algorithms (WFA) for stochastic particle coagulation, J. Comput. Phys., 230, 8427–8451, doi:10.1016/j.jcp.2011.07.027, 2011.

Devroye, L.: Non-Uniform Random Variate Generation, Springer-Verlag, New York, doi:10.1007/978-1-4613-8643-8, 1986.

Estivill-Castro, V. and Wood, D.: A Survey of Adaptive Sorting Algorithms, ACM Comput. Surv., 24, 441–476, doi:10.1145/146370.146381, http://doi.acm.org/10.1145/146370.146381, 1992.

Gillespie, D. T.: The Stochastic Coalescence Model for Cloud Droplet Growth, J. Atmos. Sci., 29, 1496–1510, doi:10.1175/1520-0469(1972)029<1496:TSCMFC>2.0.CO;2, http://dx.doi.org/10.1175/1520-0469(1972)029<1496:TSCMFC>2.0.CO;2, 1972.

Gillespie, D. T.: An Exact Method for Numerically Simulating the Stochastic Coalescence Process in a Cloud, J. Atmos. Sci., 32, 1977–1989, doi:10.1175/1520-0469(1975)032<1977:AEMFNS>2.0.CO;2, http://dx.doi.org/10.1175/1520-0469(1975)032<1977:AEMFNS>2.0.CO;2, 1975.

Golovin, A. M.: The solution of the coagulation equation for cloud droplets in a rising air current, Bull. Acad. Sci. USSR, Geophys. Ser., 5, 783–791, 1963.

Grabowski, W. W. and Wang, L.-P.: Growth of Cloud Droplets in a Turbulent Environment, Annual Review of Fluid Mechanics, 45, 293–324, doi:10.1146/annurev-fluid-011212-140750, http://dx.doi.org/10.1146/annurev-fluid-011212-140750, 2013.

Hall, W. D.: A Detailed Microphysical Model Within a Two-Dimensional Dynamic Framework: Model Description and Preliminary Results, J. Atmos. Sci., 37, 2486–2507, doi:10.1175/1520-0469(1980)037<2486:ADMMWA>2.0.CO;2, http://dx.doi.org/10.1175/1520-0469(1980)037<2486:ADMMWA>2.0.CO;2, 1980.

Heymsfield, A. and Westbrook, C.: Advances in the estimation of ice particle fall speeds using laboratory and field measurements, J. Atmos. Sci., 67, 2469–2482, 2010.

Hoffmann, F.: The Effect of Spurious Cloud Edge Supersaturations in Lagrangian Cloud Models: An Analytical and Numerical Study, Mon. Weather Rev., 144, 107–118, doi:10.1175/MWR-D-15-0234.1, 2016.

Hoffmann, F., Raasch, S., and Noh, Y.: Entrainment of aerosols and their activation in a shallow cumulus cloud studied with a coupled LCM-LES approach, Atmos. Res., 156, 43 – 57, doi:http://dx.doi.org/10.1016/j.atmosres.2014.12.008, 2015.

Kessler, E.: On distribution and continuity of water substance in atmospheric circulations, Mon. Americal Met. Soc., Boston, 10, 1–84, 1969a.

Kessler, E.: Models of microphysical parameters and processes, Meteorol. Monogr., 10, 26–31, 1969b.

Khain, A., Ovtchinnikov, M., Pinsky, M., Pokrovsky, A., and Krugliak, H.: Notes on the state-of-the-art numerical modeling of cloud microphysics, Atmos. Res., 55, 159–224, 2000.

Khairoutdinov, M. and Kogan, Y.: A new cloud physics parameterization in a large-eddy simulation model of marine stratocumulus, Mon. Wea. Rev., 128, 229–243, 2000.

Kolodko, A. and Sabelfeld, K.: Stochastic particle methods for Smoluchowski coagulation equation: Variance reduction and error estimations, Monte Carlo Methods and Applications, 9, 315–339, doi:10.1163/156939603322601950, 2003.

Kostinski, A. and Shaw, R.: Fluctuations and luck in droplet growth by coalescence, Bull. Am. Meteorol. Soc., 86, 235–244, doi:10.1175/BAMS-86-2-235, 2005.

L'Ecuyer, P. and Simard, R.: TestU01: A C Library for Empirical Testing of Random Number Generators, ACM Trans. Math. Softw., 33, doi:10.1145/1268776.1268777, http://doi.acm.org/10.1145/1268776.1268777, 2007.

Lee, J., Noh, Y., Raasch, S., Riechelmann, T., and Wang, L.-P.: Investigation of droplet dynamics in a convective cloud using a Lagrangian cloud model, Meteorology and Atmospheric Physics, 124, 1–21, doi:10.1007/s00703-014-0311-y, cited By 0, 2014.

Long, A. B.: Solutions to the Droplet Collection Equation for Polynomial Kernels, J. Atmos. Sci., 31, 1040–1052, doi:10.1175/1520-0469(1974)031<1040:STTDCE>2.0.CO;2, http://dx.doi.org/10.1175/1520-0469(1974)031<1040:STTDCE>2.0.CO;2, 1974.

Maisels, A., Einar Kruis, F., and Fissan, H.: Direct simulation Monte Carlo for simultaneous nucleation, coagulation, and surface growth in dispersed systems, Chemical Engineering Science, 59, 2231–2239, doi:10.1016/j.ces.2004.02.015, 2004.

Maronga, B., Gryschka, M., Heinze, R., Hoffmann, F., Kanani-Sühring, F., Keck, M., Ketelsen, K., Letzel, M. O., Sühring, M., and Raasch, S.: The Parallelized Large-Eddy Simulation Model (PALM) version 4.0 for atmospheric and oceanic flows: model formulation, recent developments, and future perspectives, Geosci. Model Dev., 8, 2515–2551, doi:10.5194/gmd-8-2515-2015, http://www.geosci-model-dev.net/8/2515/2015/, 2015.

Matsumoto, M. and Nishimura, T.: Mersenne Twister: a 623-dimensionally equidistributed uniform pseudo-random number generator, ACM Transactions on Modeling and Computer Simulation, 8, 3–30, 1998.

Mitchell, D.: Use of mass- and area-dimensional power laws for determining precipitation particle terminal velocities, J. Atmos. Sci., 53, 12, 1710 – 1723, 1996.

Naumann, A. K. and Seifert, A.: A Lagrangian drop model to study warm rain microphysical processes in shallow cumulus, J. Adv. Model. Earth Syst., 7, 1136–1154, 2015.

Naumann, A. K. and Seifert, A.: Recirculation and growth of raindrops in simulated shallow cumulus, Journal of Advances in Modeling Earth Systems, 8, 520–537, doi:10.1002/2016MS000631, http://dx.doi.org/10.1002/2016MS000631, 2016.

Paoli, R., Hélie, J., and Poinsot, T.: Contrail formation in aircraft wakes, J. Fluid Mech., 502, 361–373, 2004.

Paoli, R., Nybelen, L., Picot, J., and Cariolle, D.: Effects of jet/vortex interaction on contrail formation in supersaturated conditions, Phys. Fluids, 25, 1–28, 2013.

Rade, L. and Westergren, B.: Springers mathematische Formeln: Taschenbuch für Ingenieure, Naturwissenschaftler, Informatiker, Wirtschaftswissenschaftler, Springer Berlin Heidelberg, doi:10.1007/978-3-642-57239-5, 2000.

Riechelmann, T., Noh, Y., and Raasch, S.: A new method for large-eddy simulations of clouds with Lagrangian droplets including the effects of turbulent collision, New Journal of Physics, 14, 065 008, 2012.

Riemer, N., West, M., Zaveri, R. A., and Easter, R. C.: Simulating the evolution of soot mixing state with a particle-resolved aerosol model, J. Geophys. Res., 114, n/a–n/a, doi:10.1029/2008JD011073, d09202, 2009.

Schmitt, C. G. and Heymsfield, A. J.: The Dimensional Characteristics of Ice Crystal Aggregates from Fractal Geometry, J. Atmos. Sci., 67, 1605–1616, doi:10.1175/2009JAS3187.1, http://dx.doi.org/10.1175/2009JAS3187.1, 2010.

Scott, W. T.: Analytic Studies of Cloud Droplet Coalescence I, J. Atmos. Sci., 25, 54–65, doi:10.1175/1520-0469(1968)025<0054:ASOCDC>2.0.CO;2, http://dx.doi.org/10.1175/1520-0469(1968)025<0054:ASOCDC>2.0.CO;2, 1968.

Seifert, A. and Beheng, K. D.: A double-moment parameterization for simulating autoconversion, accretion and selfcollection, Atmos. Res., 59, 265–281, 2001.

Shima, S., Kusano, K., Kawano, A., Sugiyama, T., and Kawahara, S.: The super-droplet method for the numerical simulation of clouds and precipitation: a particle-based and probabilistic microphysics model coupled with a non-hydrostatic model, Q. J. R. Meteorol. Soc., 135, 1307–1320, 2009.

Shirgaonkar, A. and Lele, S.: Large Eddy Simulation of Early Stage Contrails: Effect of Atmospheric Properties, 44 th AIAA Aerospace Sciences Meeting and Exhibit, 0, 1–13, 2006.

Simmel, M., Trautmann, T., and Tetzlaff, G.: Numerical solution of the stochastic collection equation - comparison of the Linear Discrete Method with other methods, Atmos. Res., 61, 135–148, 2002.

Sölch, I. and Kärcher, B.: A large-eddy model for cirrus clouds with explicit aerosol and ice microphysics and Lagrangian ice particle tracking, Q. J. R. Meteorol. Soc., 136, 2074–2093, 2010.

Sölch, I. and Kärcher, B.: Process-oriented large-eddy simulations of a midlatitude cirrus cloud system based on observations, Q. J. R. Meteorol. Soc., 137, 374–393, 2011.

Tzivion, S., Feingold, G., and Levin, Z.: An Efficient Numerical Solution to the Stochastic Collection Equation, J. Atmos. Sci., 44, 3139–3149, doi:10.1175/1520-0469(1987)044<3139:AENSTT>2.0.CO;2, http://dx.doi.org/10.1175/1520-0469(1987)044<3139:AENSTT>2.0.CO;2, 1987.

Unterstrasser, S.: Large eddy simulation study of contrail microphysics and geometry during the vortex phase and consequences on contrail-to-cirrus transition, J. Geophys. Res., 119, 7537–7555, doi:10.1002/2013JD021418, http://dx.doi.org/10.1002/2013JD021418, 2014.

Unterstrasser, S. and Görsch, N.: Aircraft-type dependency of contrail evolution, J. Geophys. Res., 119, 14,015–14,027, doi:10.1002/2014JD022642, http://dx.doi.org/10.1002/2014JD022642, 2014JD022642, 2014.

Unterstrasser, S. and Sölch, I.: Optimisation of simulation particle number in a Lagrangian ice microphysical model, Geosci. Model Dev., 7, 695–709, doi:10.5194/gmd-7-695-2014, http://www.geosci-model-dev.net/7/695/2014/, 2014.

Unterstrasser, S., Gierens, K., Sölch, I., and Wirth, M.: Numerical simulations of homogeneously nucleated natural cirrus and contrail-cirrus. Part 2: Interaction on local scale, Meteorol. Z., doi:10.1127/metz/2016/0780, http://dx.doi.org/10.1127/metz/2016/0780, 2016.

Wacker, U. and Seifert, A.: Evolution of rain water profiles resulting from pure sedimentation: Spectral vs. parameterized description, Atmos. Res., 58, 19–39, 2001.

Wang, L.-P., Xue, Y., Ayala, O., and Grabowski, W. W.: Effects of stochastic coalescence and air turbulence on the size distribution of cloud droplets, Atmos. Res., 82, 416–432, 2006.

Wang, L.-P., Xue, Y., and Grabowski, W. W.: A bin integral method for solving the kinetic collection equation, J. Comput. Phys., 226, 59–88, 2007.

Xue, H., Feingold, G., and Stevens, B.: Aerosol effects on clouds, precipitation, and the organization of shallow cumulus convection, J. Atmos. Sci., 65, 392–406, 2008.