Reply to the Referee #2

July 14, 2017

We would like to thank the reviewers for their questions and comments. Before we answer them, we need to point out that we found an inconsistency in the way the collision efficiency tables were interpolated in the SDM and in the Bott method. It only affected simulations that use efficiencies from Hall (1980) for large droplets and from Davis (1972) for small droplets. This kind of collision kernel was used in Sections 5 and 6. The SDM simulations presented in Figs. 7, 8 and 9 were repeated with the problem fixed. The main difference is that now we see convergence of the "one-to-one" results to the Smoluchowski equation in the slow-coalescence case. What did not change is the fact that using larger coalescence cells can decrease the rate of conversion of cloud to rain drops due to additional collisions between rain drops. In consequence, using the Smoluchowski equation can underestimate the amount of rain produced. The problem affected only large drops, with radius greater than ca. 90 microns. Therefore the lucky droplet analysis from Sec. 6, in which droplets grow only up to 40 microns, remains valid.

Answer to the Anonymous Referee #2.

- The presentation is too short to fully understand what has been done and to be able to judge it. Moreover, the statements are too general. Also the motivation behind choosing the presented tests must be made clearer. In the present style the presented tests resemble a bit a random collection of tests related to a specific SD coalescence algorithm.

We have made an effort to make the statements and the presentation more specific. Regarding the motivation for different tests, we have rewritten the last paragraph to the introduction to make it more clear:

" The Shima algorithm is not based on the Smoluchowski equation, but, similarly to the master equation, on the assumption that the volume is well-mixed. The Shima algorithm introduces some simplifications that may increase the scale of fluctuations in the number of collisions, as described in Sec. 2.

These simplifications are not necessary in the limiting case of a single computational particle representing a single real particle, what we call "one-to-one" simulations. Then, the Shima algorithm should be equivalent to the SSA, i.e. it should produce a realization in agreement with the master equation. To show that this is true, we compare the Shima algorithm with the master equation and the SSA in Sec. 3. We also compare it with the more fundamental DNS approach in Sec. 4. Once the "one-to-one" approach is shown to be at the same level of precision as the master equation, we use it to study some physical processes that are related to the stochastic nature of coalescence. The way the sol-gel transition time changes with system size is studied in Sec . 3 and in Sec. 6, we quantify how quickly the luckiest cloud droplets become rain drops. In addition, we use the "one-to-one" approach to validate more approximate methods. The Shima algorithm with multiplicities greater than one is studied in Sec. 4. We determine how many computational particle s are required to obtain the correct mean autoconversion time and correct fluctuations in the autoconversion time. Next, in Sec. 5, we determine how large the system has to be for the Smoluchowski equation to correctly represent the rate of rain formation. Throught the paper we observe that evolution of the system strongly depends on its size. The size of a well-mixed air parcel is estimated in Sec. 7 and some implications for cloud simulations are discussed in Sec. 8. "

In addition, we changed the titles of Secs. 3 and 4 to make it more clear what is their purpose.

General points 1. The description of your method and results is too short and often not clear enough. For a reviewer it is difficult to understand what you have done in detail and this makes it hard to thoroughly rate your work. The description in section 2 must be considerably expanded and be more precise. Also the results section should be improved. For example the motivation behind showing the comparison in section 3 is not really clear to me. As an example I go over page 3 and try to highlight what I miss there and where I suspect that other readers would get stuck:

The method was described in detail by Shima et al. (2009), so our intention was only to describe how our simulation method differs from theirs. Apparently this makes the method not clear, so we extend the description of the method as asked by the Reviewer in the following points. Regarding the motivation behind different comparisons, it is now given in the Introduction, as explained in the answer to the previous comment.

It is not always clear if you talk about real droplets or super-droplets. It should be always clear if your statement relates to the real world or the super droplet world.

It is now clarified in Sec. 1:

" We will refer to these computational particles as super-droplets (SDs). The words "droplets" and "drops" are reserved for real hydrometeors. "

You should define the coalescence probability of a SD pair.

It is now given in Sec. 2: " Probability of coalescence of two SDs i and j that belong to the same collision pair is $P_{SD}(r_i, r_j, \xi_i, \xi_j)$ =max$(\xi_i, \xi_j)P(r_i, r_j)(N_{SD}(N_{SD}-1)/2)/\lfloor N_{SD}/2 \rfloor$ (Shima et al., 2009).
"

Can collections occur between droplets in one SD?

We use gravitational coalescence kernels, so droplets in one SD all have the same terminal velocity and therefore cannot collide. We now say it explicitly in Sec. 2:

" Real droplets represented by the same SD cannot collide with each other, because they have the same sedimentation velocities. "

l.9: Do you use two types of simulations or is it only the initialisation that differs between the two types? Then I would not call it types of simulations.

We use two types of simulations. The main difference, besides the initialisation, is that in the "one-to-one" simulations the timestep is adaptive, as stated in Sec. 2:

"Timestep length is adapted at each step to ensure that none of the collision pairs has coalescence probability greater than one."

In the "constant SD" simulations the timestep is constant and multiple collisions between SDs in a single timestep are allowed, as said in Sec. 2:

" In this type of simulation, the time step length is constant $\Delta t = 1$ s. It is not adapted, as it is done in the "one-to-one" simulations, to make the simulation computationally more efficient. Using constant time step length can make the coalescence probability exceed unity. If it does, it represent multiple collisions between a pair of SDs (Shima et al., 2009). "

In general I can imagine how the constant SD initialisation works, however I am confused by your description. o l.21: do you mean large or small NSD? Or large r?

As written, we mean large N_{SD}. For large N_{SD}, Δl_r becomes small and therefore r_{max} is small.

o rmin and rmax are defined by the relation in line 18. Using the ln(r) definition of line 19, you end up with implicit equations in the two variables. I am pretty sure this is not what you wanted to say.

Yes, we end up with implicit equations for these variables. Many pairs of values of r_{min} and r_{max} could satisfy them. We find our solution numerically, what is now explained in more detail in Sec. 2:

" The first step of the initialization is finding the largest and smallest initial super-droplet radius, r_{max} and r_{min}. They are found iteratively, starting with $r_{min} = 10^{-9}$ m and $r_{max} = 10^{-3}$ m. We require that they satisfy the condition

$$n(\ln(r_e))\Delta l_r V \geq 1, \tag{1}$$

where r_e is either r_{max} or r_{min}, $n(\ln(r))$ is the initial droplet size distribution and $\Delta l_r = (\ln(r_{max}) - \ln(r_{min}))/N_{SD}$. In each iteration, if r_{min} (r_{max}) does not satisfy (1), it is increased (decreased) by 1%. "

o You initialise additional SDs to better represent the tails of the SDs. Adding those SDs to your ensemble, isnt it necessary to reduce the multiplicity of the SD drawn from the largest bin?

No, in "constant SD" the right edge of the largest bin is r_{max}, so effectively the distribution is cut at r_{max} and droplets with $r > r_{max}$, that would be present in the real system, are not accounted for. Adding additional SDs with $r > r_{max}$ fixes this problem and does not affect the number of droplets with $r \leq r_{max}$ (i.e. the multiplicity of SDs with $r \leq r_{max}$).

Is the refinement done at both sides?

No, only on the large radius side. We now write explicitly in Sec. 2:

" We do not add SDs from the small tail of the distribution, because very small droplets are of little importance for rain formation. "

In the end, the actual number of SDs should be higher than the nominal value NSD? How much higher?

Yes, it is a little higher, what is now written in Sec. 2:

" This makes the actual number of SDs higher than the prescribed value N_{SD}, typically by ca. 1%. "

Is the one-to-one initialisation the limiting case of the constant SD initialisation?

No, because using $N_{SD} = N_0$ (i.e. multiplicity = 1) in a "constant SD" simulation would result in relatively small r_{max} and large r_{min}. Then this type of initialisation would not represent well the given distribution. Therefore a different approach to initialisation is used in the "one-to-one" simulations.

Are the multiplicities in the constant SD approach integer values? How is the rounding done?

Yes they are, as explained in Shima et al. (2009). The rounding is done to the nearest integer. The error introduced is small, because multiplicities are high.

o **l.20: To be more explicit, you should write that one SD is created per bin and its radius is randomly selected.**

It is written in Sec. 2:
" Once r_{min} and r_{max} are found, radius of one SD is randomly selected within each bin of size Δl_r. "

o **l.23: Do you want to say that the probability is allowed to exceed 1. Why does this occur only here, and not in the one-to-one simulation? Is the time step longer? Why dont you reduce it then in this case?**

Yes, in the method of Shima et al. (2009) the coalescence probability is allowed to exceed 1. It is a consequence of keepeing the time step length constant. We added to Sec. 2:
" In this type of simulation, the time step length is constant $\Delta t = 1$ s. It is not adapted, as it is done in the "one-to-one" simulations, to make the simulation computationally more efficient. Using constant time step length can make the coalescence probability exceed unity. If it does, it represent multiple collisions between a pair of SDs (Shima et al., 2009). "

o **What are multiple collections?**

It means that, during a single time step, a pair of SDs collides more than once. The details are given in Shima (2009).

o **Do you use a constant time step throughout a simulation?**

In the "one-to-one" simulations it is adaptive, as said in Sec. 2:
" Time step length is adapted at each time step to ensure that none of the collision pairs has coalescence probability greater than one. "
In the "constant SD" simulations it is constant, what is now explicitly stated in Sec. (2):
" In this type of simulations, the time step length is constant $\Delta t = 1$ s. "

2. Your conclusions are too general. Your paper reads like a general comparison between the one and only SDM approach and all kinds of other algorithms. Coalescence (or similarly coagulation of aerosols and dust or aggregation of ice crystals) can be treated in many ways in particle-based approaches (see algorithms by Andrejczuk et al, Shima et al, Riechelmann et al, Solch & Karcher, Zsom & Dullemond, Kolodko & Sabelfeld, deVille et al). There is

not the one natural way to do this. Similarly, various algorithms exist for bin model approaches. Please make clearer which statements hold in general and which are probably only valid for your specific SDM.

Regarding coalescence, the main difference between these particle-based approaches (with the exception of the DeVille algorithm, which is based on the Smoluchowski equation) is in what is the outcome of a collision of super-droplets with multiplicities $\xi > 1$. Majority of our simulations were the "one-to-one" simulations, in which $\xi = 1$. In that case, it is straightforward what the result of a collision should be, so differences between these algorithms disappear. Moreover, like the master equation, the "one-to-one" simulations are only based on the assumption that the cell is well-mixed. The numerical trick of reducing the number of collision pairs ("linear sampling") does not affect the fluctuations, as we show in Sec. 3. Therefore the "one-to-one" simulations are quite similar to the SSA. They are at the same level of accuracy as the master equation: less precise than the DNS, more precise than the Smoluchowski equation. It is now cleary stated in the Introduction:

" The Shima algorithm is not based on the Smoluchowski equation, but, similarly to the master equation, on the assumption that the volume is well-mixed. The Shima algorithm introduces some simplifications that may increase the scale of fluctuations in the number of collisions, as described in Sec. 2. These simplifications are not necessary in the limiting case of a single computational particle representing a single real particle, what we call "one-to-one" simulations. Then, the Shima algorithm should be equivalent to the SSA, i.e. it should produce a realization in agreement with the master equation. To show that this is true, we compare the Shima algorithm with the master equation and the SSA in Sec. 3. We also compare it with the more fundamental DNS approach in Sec. 4. "

In simulations with $\xi > 1$, we use the Shima method, as it was found to be optimal by Unterstrasser et al. (2017) and by Li et al. (2017). These simulations are used only in Sec. 4 in order to determine their accuracy, as explained by an added paragraph in the introduction:

" In addition, we use the "one-to-one" approach to validate more approximate methods. The Shima algorithm with multiplicities greater than one is studied in Sec. 4. We determine how many computational particles are required to obtain the correct mean autoconversion time and correct fluctuations in the autoconversion time. "

A recent study by Unterstrasser et al compared three different SD algorithms for the collection process (your all-or-nothing SDM is among the tested algorithms). Each algorithm has its strengths and weaknesses. One major finding was that the performance depends strongly on how the SDs are initialised from a given size distribution. This has several implications: 1. an accurate description of your initialisation is needed (see point 1) and

Unterstrasser et al. (2017) found that the Shima method is optimal, what we now say in the introduction: " A thorough comparison of coalescence algorithms from Lagrangian methods was done by Unterstrasser et al. (2017). It lead to the conclusion that the method of Shima (2009) "yields the best results and is the only algorithm that can cope with all tested kernels". It was also found to be optimal in DNS tests Li et al. (2017). In the light of these results, we choose to use the coalescence algorithm of Shima (2009) in this work. "

We also observed that the way the initialisation is done is important. We believe that now our initialisation algorithm is described more clearly.

2.Are your simulations sensitive to initialisation details besides the number of SDs?

It is only sensitive to the number of SDs. Initial values of r_{min} and r_{max}, if reasonable, do not affect it much.

3. In my opinion you present five types of tests: 1. Comparisons with algorithms for the master equation (ME), 2. Comparisons with DNS algorithms 3. Comparisons with algorithms for the Smoluchowski equation (KCE) 4. Comparison with analytical results and 5. Stand-alone sensitivity tests of your SD algorithm. With each type of test different aspects of your SD method can be tested. Each reference model you compare with (Bott, Alfonso, Onishi, Kostinski & Shaw) differs in which physical processes are explicitly treated. You often miss to clearly specify the purpose of and the motivation behind the individual evaluation steps you take. This must be made clearer in the revised version.

We believe that it is now made clear by the following paragraph in the introduction:

" The Shima algorithm is not based on the Smoluchowski equation, but, similarly to the master equation, on the assumption that the volume is well-mixed. The Shima algorithm introduces some more simplifications that may increase the scale of fluctuations in the number of collisions, as described in Sec. 2. These additional simplifications are not necessary in the limiting case of a single computational particle representing a single real particle, what we call "one-to-one" simulations. Then, the Shima algorithm should be equivalent to the SSA, i.e. it should produce a trajectory in agreement with the master equation. To show that this is true, we compare the Shima algorithm with the master equation and the SSA in Sec. 3. We also compare it with the more fundamental DNS approach in Sec. 4. Once the "one-to-one" approach is shown to be at the same level of precision as the master equation, we use it to study some physical processes that are related to the stochastic nature of coalescence. The way the sol-gel transition time changes with system size is studied in Sec. 3 and in Sec. 6, we quantify how quickly the luckiest cloud droplets become rain drops. In addition, we use the "one-to-one" approach to validate more

approximate methods. The Shima algorithm with multiplicities greater than one is studied in Sec. 4. We determine how many computational particles are required to obtain the correct mean autoconversion time and correct fluctuations in the autoconversion time. Next, in Sec. 5, we determine how large the system has to be for the Smoluchowski equation to correctly represent the rate of rain formation. "

4. Your study is written in a style that implies that your SD algorithm does not solve the KCE. By comparing your SD results with KCE results you seemingly disclose shortcomings of the KCE description. In fact, SDMs also rely on KCE and your study is not really suited to question the physics of the various formulations. In my understanding, you compare different numerical strategies of solving the KCE.

The SD algorithm is not a method of solving the KCE. Contrary to the KCE, it does include correlations between number of droplets of different sizes. The "one-to-one" simulations are similar to the SSA, i.e. they produce a single trajectory that follows the master equation (c.f. Sec. 2). As such, they are more precise than the KCE, so they are well-suited to discolse shortcomings of the KCE. To us it is not clear how the "constant SD" simulations relate to the KCE and the master equation. They have been shown to give mean result in agreement with the KCE in large systems (Shima et al., 2009, Unterstrasser et al., 2016). Regarding fluctuations, using the "all-or-nothing" algorithm should amplify fluctuations, because it introduces unrealistic correlations between number of droplets of different sizes. In Sec. 4 we quantify how much the fluctuations amplitude increases.

The probabilistic nature of your SDM is inherent to your specific algorithm. The SDMs of Riechelmann or Andrejczuk are not probabilistic. On the other hand, I agree that the probabilistic component of your SDM looks like it attempts to mimic the processes in nature. Nevertheless, the two should not be mistaken. In the superdropelt world of your SDM, the probabilistic approach comes from the fact, that each superdropelt represents a large number of real droplets and you may or may not perform a superdroplet collection. This not per se related to the stochastic nature of the real world processes that accounted for in the master equation.

The probabilistic nature of our SDM has the same source as the probabilistic nature of the master equation, i.e. the fact that a collision between a pair of SDs happens with some probability, according to the assumption that the volume is well-mixed. To our knowledge, this Monte Carlo approach to collisions is used in all SDMs, including the ones of Riechelmann and Andrejczuk. If multiplicities are equal to one ("one-to-one" simulations), the SDM is as much related to the real world process as the master equation. If multiplicities are greater than one,

various SDM algorithms start to differ. In the Shima algorithm that we use, the scale of fluctuations is increased, because the number of collision trials is lower than it would be in reality. We try to quantify how much it is increased. In the Riechelmann and Andrejczuk SDMs, the fluctuations are lower than in the one of Shima. Unfortunately, they do not give mean results as good as the Shima algorithm (Unterstrasser et al., 2016).

Major points: 5. In my understanding there two mathematical descriptions of the coalescence process: The Smoluchowski equation and the master equation where only the latter accounts for correlations. You never state it explicitly but I suppose that your SD method is based on solving KCE. Many spectral bin methods exist for the KCE and in the recent past SD methods for KCE were introduced. Hence, demonstrating agreement between your KCE solver and the ME solver by Alfonso is only reasonable for a physical problem where correlations are negligible. If you want to show the numerical consistency of your approach, then a comparison with other well-established KCE solvers would be more straightforward. Could you give reasons why you compare results of your KCE solver and an ME solver? Or do you want to claim that SDMs implicitly account for correlations?

Our SD method is not based on solving KCE. The "one-to-one" simulations are at the level of precision of the master equation, what is now explicitly written in the introduction:

" The Shima algorithm is not based on the Smoluchowski equation, but, similarly to the master equation, on the assumption that the volume is well-mixed. The Shima algorithm introduces some more simplifications that may increase the scale of fluctuations in the number of collisions, as described in Sec. 2. These additional simplifications are not necessary in the limiting case of a single computational particle representing a single real particle, what we call "one-to-one" simulations. Then, the Shima algorithm should be equivalent to the SSA, i.e. it should produce a trajectory in agreement with the master equation. To show that this is true, we compare the Shima algorithm with the master equation and the SSA in Sec. 3. "

We compare "one-to-one" simulations with the master equation to validate the claim that they are at the same level of precision, i.e. that "one-to-one" method accounts for correlations. In the problem of Alfonso, correlations are very important and, as shown in Alfonso and Raga (2017), the KCE does not solve it well.

6. I am not exactly sure what you intended to show in section 3. Did you want to demonstrate the suitability of your linear sampling of droplet pairs? Then you should compare results of your algorithm, once with linear sampling and once considering all possible combination. For a simulation with 30 droplets (as done in the comparison

with Alfonso) this should be feasible. How do the curves in Fig.3 and 4 change, if you evaluate all possible combinations instead of linear sampling (at least for N0=10 and 100)?

We intended to show that "one-to-one" simulations agree with the master equation. Linear sampling is an optimization technique that we expected might be responsible for some differences between the master equation and "one-to-one" simulations. To make the comparison more detailed, now in Section 3 we compare "one-to-one" simulations, with and without linear sampling, with the master equation. We find that linear sampling does not affect mean number of collisions, nor the fluctuations in the number of collisions. Figs. 3 and 4 (up to N0=100) do not change if linear sampling is not used.

The second paragraph of Section 3 has been rewritten to explain these new results of simulations without linear sampling.

7. Is it all reasonable trying to sample a continuous exponential distribution with 10 SDs? Are the total mass and number of the SDs (averaged over all ensembles) equal to the prescribed values? What about the higher moments, do they match the values of the analytical distribution? This is an important aspect as much of your evaluated variability may come from the initialisation and not so much from stochasticity of the SDM.

Averaged over the ensamble, up to the 4-th moment of the distribution is in agreement with the prescribed one. It is true that the initial distributions can be very different between realisations and it may be the cause of large variability. For this reason we removed the $N_0 = 10$ case from Fig. 3.

8. Even though Fig.4 shows simulations with an initialisation analogous to Onishi, the presented test is a stand-alone test (following the above categorisation). Hence, further tests with other kernels or other initial size distributions can be made (as no comparison with Onishi is required) in order to corroborate your findings about how many SDs are necessary to reach convergence.

Now in Fig. 4 we compare our results with Onishi's results. Our expectation is that for other kernels relevant for cloud physics the results would not be much different. Nevertheless, we agree that tests with other kerneles, and with other initial conditions, would be useful. Such tests could easily fill a whole new paper. Our result can be considered as a guideline for users, who should do convergence tests for the specific kernels they use.

9. In Fig.4 and following figures, does the Smoluchoswki line depend on the parameter choices of the underlying Bott algorithm (time step or number of bins)? Bott is known to be diffusive; this may explain the faster generation of large droplets. Anyway I would

not call it the Smoluchowski line, it is the Bott line. The line may look differently for other bin KCE algorithms. Moreover, your lines are also Smoluchowski lines (see general point 4).

We have done convergence tests of the Bott algorithm. It is now explained in Sec. 4:

" In the Bott algorithm, we used $\Delta t = 1$ s and mass bin spacing $m_{i+1} = 2^{1/10} m_i$. The same parameters were used in each simulation presented in this manuscript. Convergence tests were done for each case. "

We agree that the Bott algorithm produces rain too soon most probably due to numerical diffusion. We now write in Sec. 4:

" The "one-to-one" results converge with increasing system volume (i.e. increasing N_0) to a value higher than the Smoluchowski result. The difference is probably caused by the numerical diffusion of the Bott algorithm. "

Labels on figures are one of: DNS, master equation, SSA, SDM ("one-to-one" or "constant SD"), Smoluchowski equation. In our view these are different appproches to solving droplet coalescence, not different numerical methods for solving some equation. The numerical methods used, i.e. the Bott algorithm for the Smoluchowski equation and the Alfonso algorithm for the master equation, are explained in text.

Minor points: 10. p.3, l21: Is the sol-gel transition an important aspect in cloud physics? You note that mass conservation is not guaranteed for some kernels. Are those kernels relevant in cloud physics? In super-droplet approaches mass conservation should be guaranteed by construction.

Mass is not conserved for the multiplicative kernel, which is not relevant in cloud physics. Nevertheless, the paper Alfonso and Raga (ACP, 2017) is a detailed study of the sol-gel transition in a small cloud volume. We decided that it will be interesting to extend their results to more realistic cases.

11. I do not understand the statement the number of collision pairs is reduced in p.4 l.4.

It was supposed to mean that linear sampling is used, i.e. less collision pairs are considered than in an exact description. Now in Sec. 2 we define the meaning of linear sampling, which is later used in the paper:

" The second simplification, that we will refer to as linear sampling, is that instead of considering all $N_{SD}(N_{SD}1)/2$ collision pairs, only $[N_{SD}/2]$ non-overlapping pairs are randomly selected. "

12. The paragraph starting p.4 l.11 should be moved to section 2.

In the paragraph it was shown that in the "one-to-one" method with linear sampling, the probability of collision between any two real droplets is the same

as in simulations without linear sampling. This does not ensure that fluctuations in the number of collisions are also correctly represented. Therefore we removed the paragraph. In its place we added to Figs. 1 and 2 the results of simulations without linear sampling, i.e. with all collision pairs considered. Their agreement with the linear sampling simulations implies that linear sampling does not affect the scale of fluctuations. Proving that was the point of the removed paragraph.

13. For Figs.1 & 2 it would be ideal to obtain the data from Alfonso and include it in the plots.

We did that.

14. P.2 l.13: Tanaka and Nakazawa present solutions for kernels other than the constant kernel. See also Table 2 in Alfonso, 2015.

We now cite Tanaka and Nakazawa in the introduction:

" The master equation was analytically solved only for monodisperse initial conditions with simple coalescence kernels (Bayewitz et al., 1974; Tanaka and Nakazawa, 1993). "

Table 2 in Alfonso (2013) gives solutions of the Smoluchowski equation, not the master equation.

15. P.5 l.5. Isnt this statement trivial? Probably any algorithm for KCE is faster than solving the master equation. A comparison among various KCE solvers would be fairer.

As explained previously, SDM is similar to the SSA and not to KCE solvers. For this reason we compare with the SSA and a solver of the master equation.

16. Figs. 2&3 show relative standard deviation of the largest droplet mass. Wouldnt it be interesting to know how large the largest droplet mass is? How many collections have occurred to form the largest droplet?

Relative standard deviation of the largest droplet mass is interesting as a measure of the sol-gel transition. We do not see a reason to show the mass of it or the number of collisions that lead to it.

17. P.7 l.10-12: I am confused. DNS simulations compute droplet trajectories and directly evaluate if there are collisions between droplets. Why do you need a coalescence kernel in this numerical approach?

It was our error. The DNS was done not for different coalescence kernels, but for different turbulence strength. We changed that sentence to:

" Small discrepancies are probably caused by the fact that the DNS included turbulence of various strength for different N_0. "

18. Fig.8: Do you also use the 20um cut-off in the Bott simulation? It is not explicitly stated in the text.

Yes, we do use it in the Bott simulation. It is now said explicitly:
" In addition, we cut the distribution to 0 at $r = 20$ m. This cutoff is used in SDM modelling as well as when solving the Smoluchowski equation. "

Technical points: Figure caption should include the information t=2500s.

We add it to the caption of Fig. 1.

P.5 l.9: droplet, not dropeltS.

Fixed.