Performance comparison among three Monte Carlo schemes for collision-coalescence: O'Rourke method, No-time counter method, and Super-droplet method

Shin-ichiro Shima (U Hyogo), Hiroshi Yamaguchi (IDAJ Co., LTD.), Hitoshi Hongou (Mazda Motor Corporation), Hideaki Yokohata (Mazda Motor Corporation)

Apr. 16, 2019

Workshop on Eulerian vs. Lagrangian methods for cloud microphysics

Abstract

Super-Droplet Method (SDM) is a Monte Carlo scheme for stochastic collision-coalescence of particles (SS et al. 2009)

Several types of Mote Carlo schemes, e.g., O'Rourke method, and No-Time Counter (NTC) method

The performance of the three are compared

It is confirmed that SDM outperforms the other two

(Efficient way of initializing super-particles is also discussed)

Contents

1. Introduction

Stochastic Collision-Coalescence of Particles

Assume a particle is characterized by d number of attributes

$\boldsymbol{a}(t)=\{a^{(1)}(t), a^{(2)}(t), \ldots, a^{(d)}(t)\}$

Assuming that the particles are well-mixed by the turbulence, we can regard collision-coalescence is a stochastic event

$$
\begin{aligned}
P_{jk} &= K(\boldsymbol{a}_j, \boldsymbol{a}_k)\frac{\Delta t}{\Delta V} \\
&= \text{probability that droplet } j \text{ and } k \\
&\quad \text{inside a small region } \Delta V \text{ will collide} \\
&\quad \text{in a short time interval } (t, t+\Delta t).
\end{aligned}
$$

All the particle pairs in ΔV can coalesce

K is called collision-coalescence kernel

Particle-Based Algorithms

- Deterministic: e.g., Andrejczuk et al. 2010; Riechelmann, Noh et al. 2012.
- Probabilistic: e.g.,
  - SDM (S.S. et al. 2009)
  - All-Or-Nothing (AON) (Sölch and Kärcher 2010, Unterstrasser et al. 2017)
  - O'Rourke (1981): for spray combustion. equiv to AON?
  - No-Time Counter (NTC) for coalescence. (Schmidt and Rutland 2000): for spray combustion
  - Weighted Flow Algorithm (DeVille et al. 2011): for aerosol dynamics. Implemented on PartMC (Riemer and West)
  - Zsom and Dullemond (2008): in astrophysics area

Performance Comparison

- Unterstrasser et al. (2017) elucidated that AON (= O'Rourke?) is more efficient than the two deterministic schemes
- Dziekan and Pawlowska (2017) pointed out SDM is much faster than AON, but no quantitative comparison
- Li et al. (2017) confirmed that SDM is better than Zsom and Dullemond's model

Objective of This Study

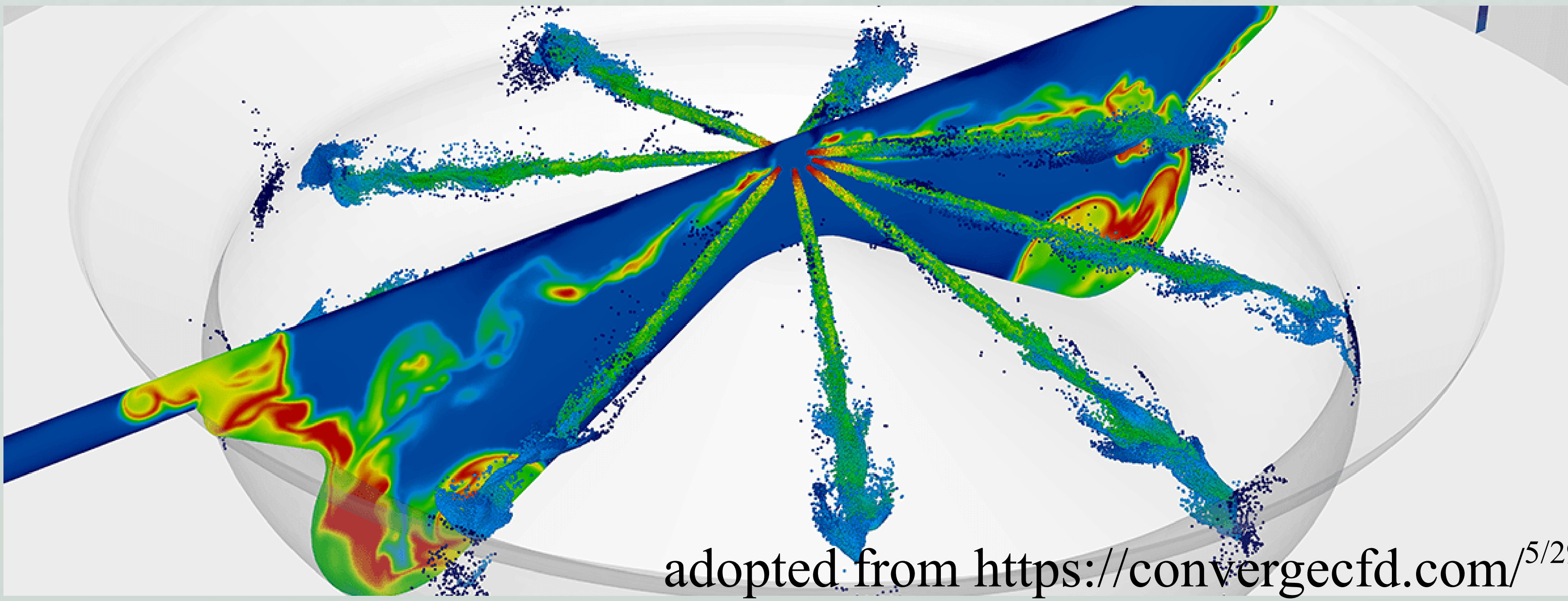
Compare the perf. of O'Rourke (=AON?), NTC, and SDM

Software

CONVERGE is a commercial CFD software:

We implemented SDM to CONVERGE as a UDF

We fixed some bugs of built-in O'Rourke and NTC



adopted from https://convergecfd.com/

2. Three Monte Carlo Schmes

Super-Particle (SP) (concept commonly used in 3 schemes)

Each SP represents multiple number of real particles (RPs), which is denoted by multiplicity ξ

Approximate RP population $\{\boldsymbol{a}_i(t)|i=1,2,\ldots,N_r(t)\}$ by SD population $\{(\xi_i(t),\boldsymbol{a}_i(t))|i=1,2,\ldots,N(t)\}$

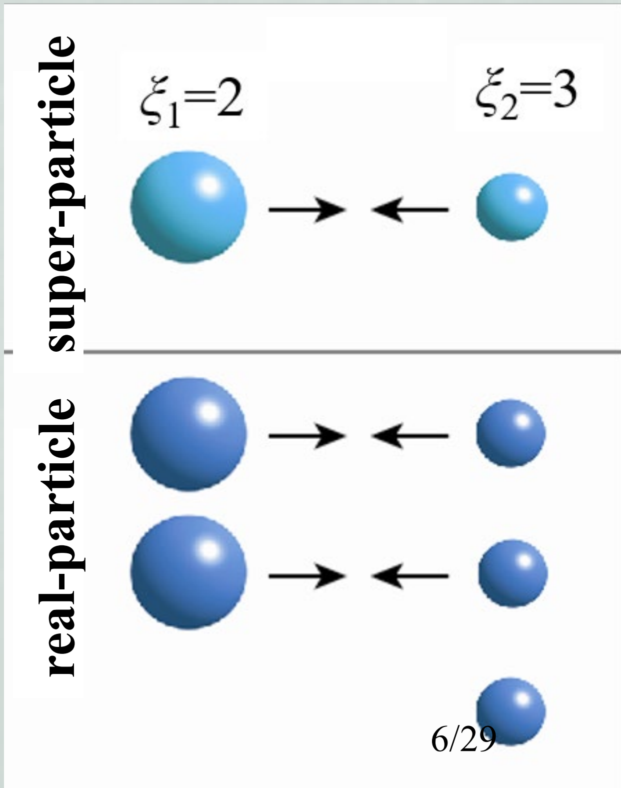
Stochastic Coalescence of SPs

When a SP pair (j,k) coalesce, we consider that $\min(\xi_j,\xi_k)$ of RPs coalesce

→ Conservation of SP num

Requiring that the expectation becomes consistent

$$P_{jk}^{(s)} := \max(\xi_j, \xi_k) P_{jk}$$



The Three Monte Carlo Schemes

O'Rourke method (O'Rourke 1981):

- Check all the SP pairs ${}_N C_2$. Cost $O(N^2)$
- Allow multiple coalescence

NTC (No-Time Counter) (Schmidt and Rutland 2000):

- Check ${}_N C_2 \cdot P^{(s)}_{max}$ SP pairs. Cost $O(N)$?
- By definition, no multiple coalescence occurs
- Parallelization is difficult due to dependence of pairs

SDM (Super-Droplet Method) (S.S. et al. 2009):

- Check $[N/2]$ SP pairs. Cost $O(N)$.
- Allow multiple coalescence
- Parallelization is easy due to the pairs are independent

3. Design of the Numerical Experiment

Experimental condition (0D simulation)

Droplets are floating in a cube of 100m on each side

They are well-mixed and coalesce repeatedly

Assume terminal speed (but do not go out of the cube)

We check evolution of droplet number and size distribution

Initial size distribution of droplets

Liquid water content: $1g/m^3$

χ^2(3.5?) distribution (CONVERGE built-in)

Sauter mean diameter (SMD): 6.1062e-5m

Initialization of SPs

CONVERGE built-in. Masses of SPs are the same.

(Note the performance is sensitive to how SPs are initialized)

Control Parameters

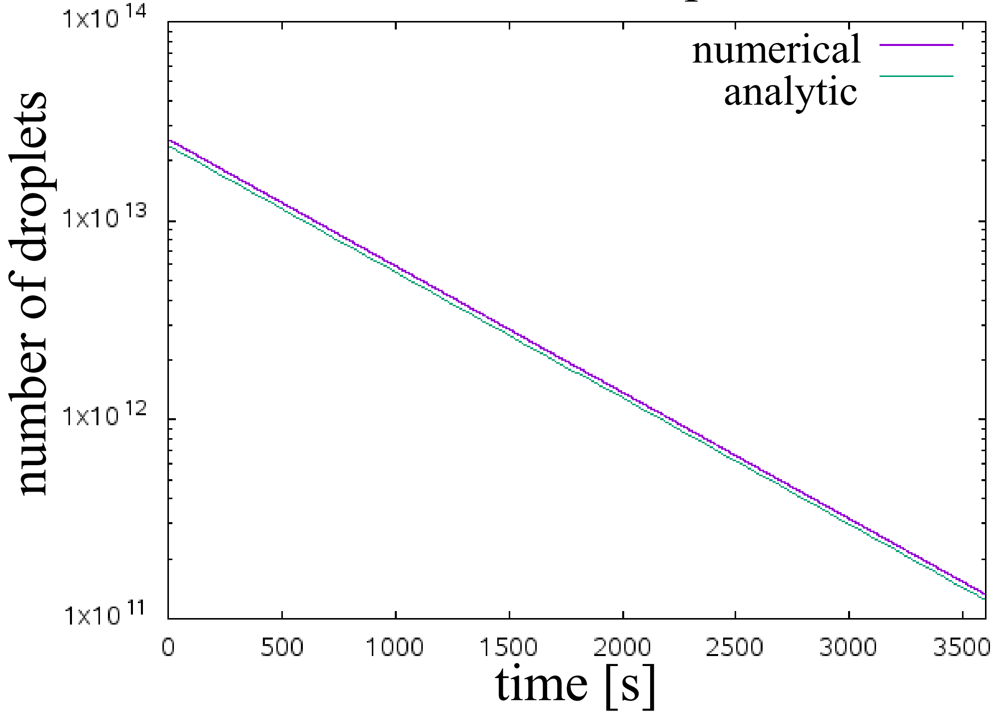
We try the combination of the following:

Collision-coalescence kernel	Golovin, geometric
Numerical scheme	SDM, NTC, O’Rourke
Num of SPs N	80, 800, 8000, 80000
dt[s]	10.0, 1.0, 0.1

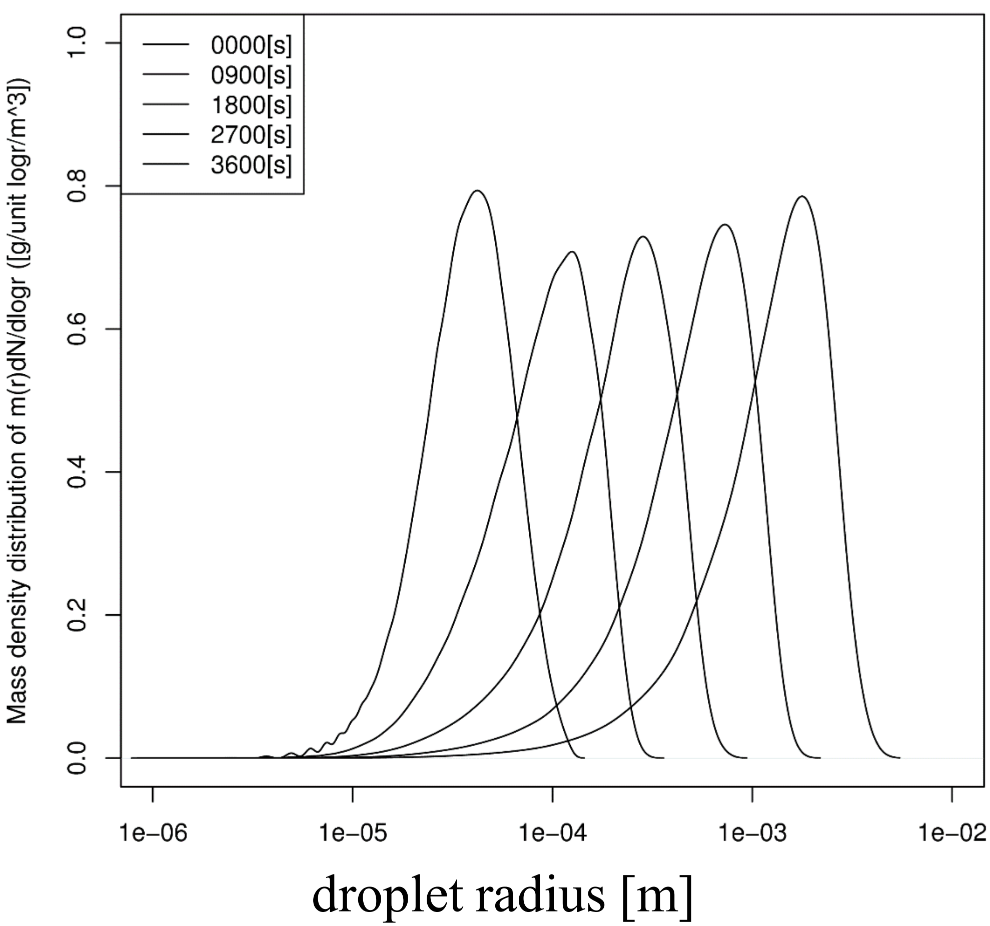
4. Result 1 (Golovin kernel)

Correct behavior (SDM, N=80000, dt=0.1s)

Time evolution of droplet number



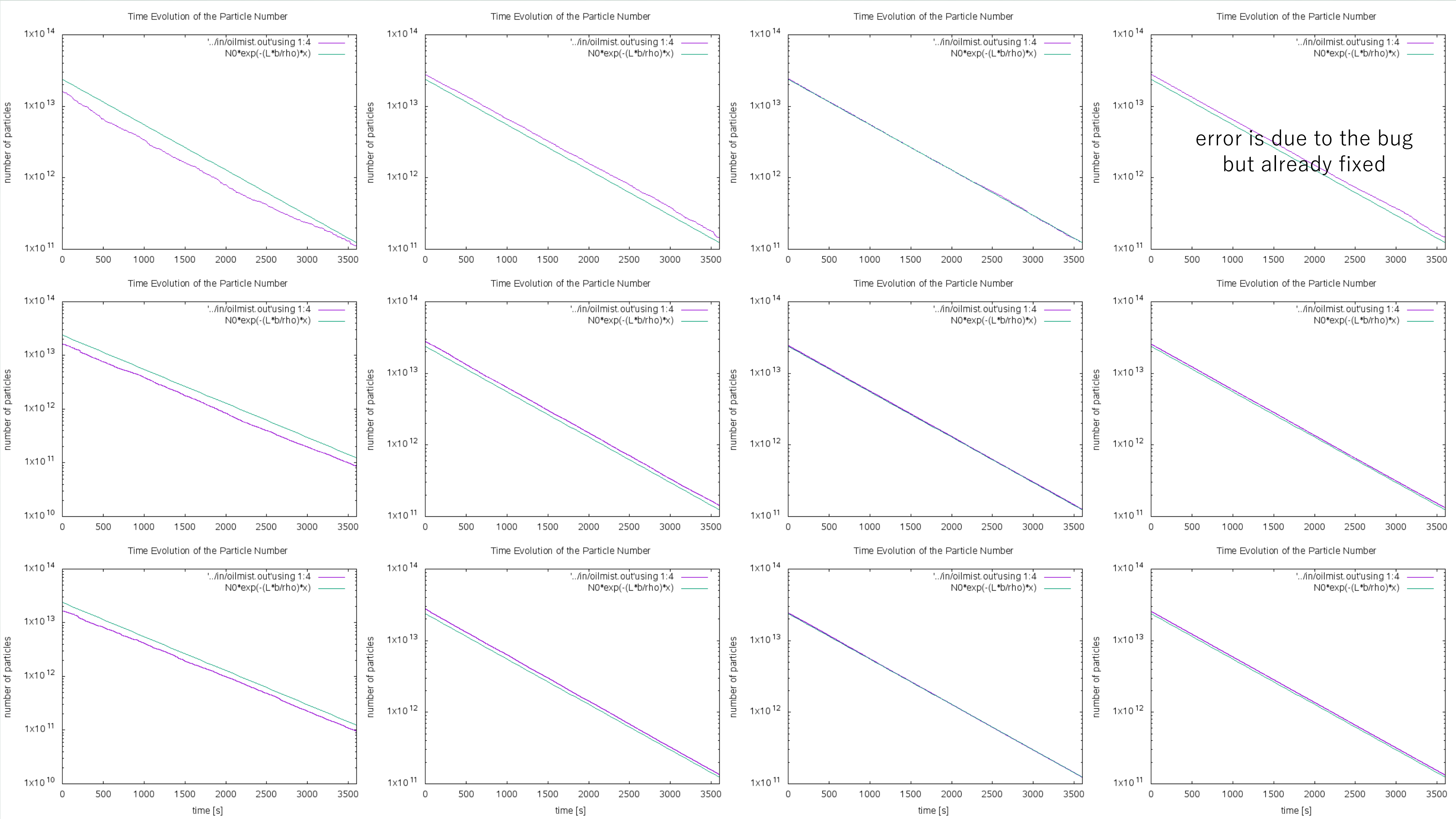
Time evolution of the mass density



Time Evolution of droplet numbers (SDM, Golovin)

N=80 | N=800 | N=8000 | N=80000

dt=10s | dt=1s | dt=0.1s



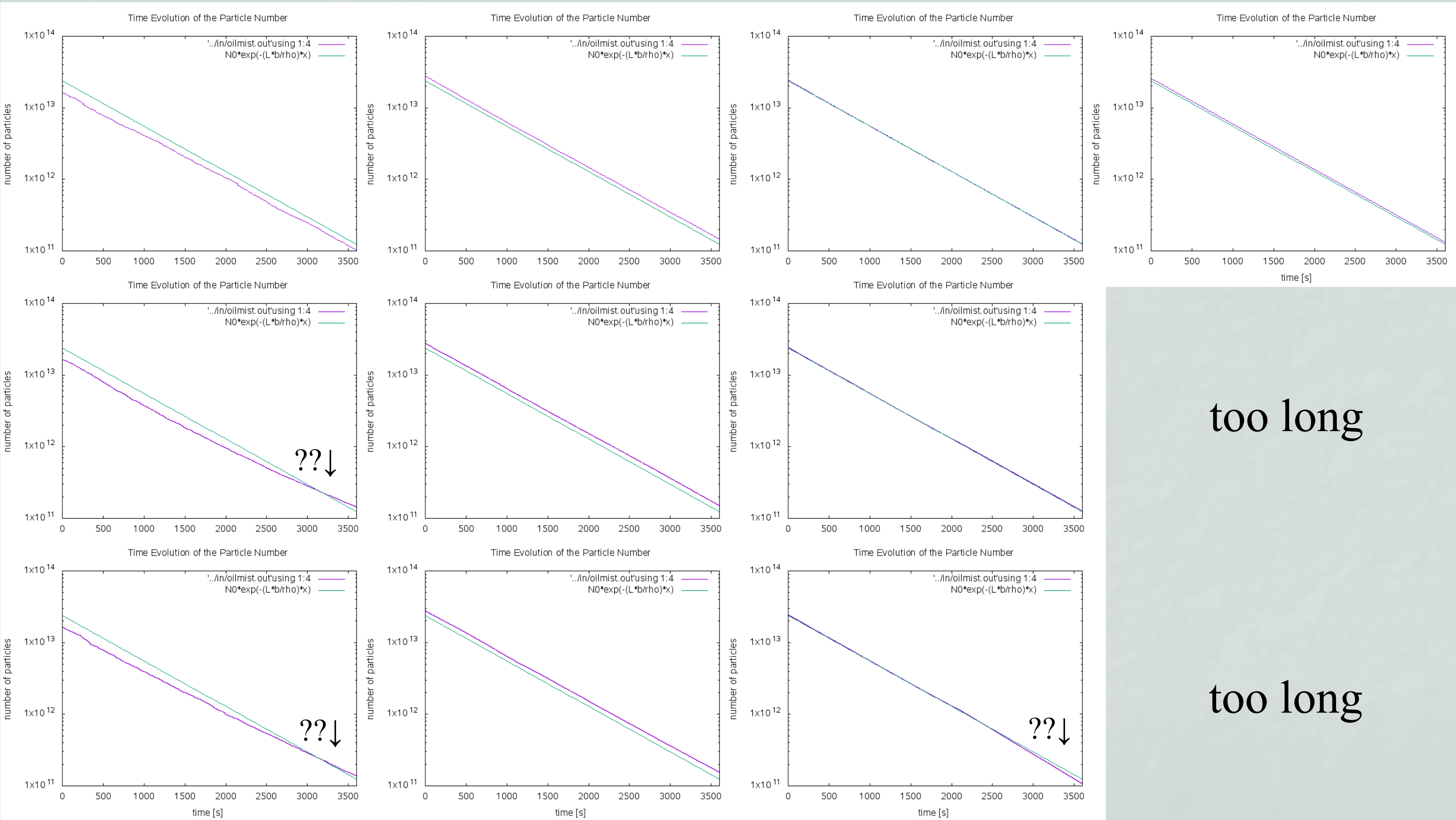
Time Evolution of droplet numbers (NTC, Golovin)

N=80 | N=800 | N=8000 | N=80000

dt=10s

dt=1s

dt=0.1s



too long

too long

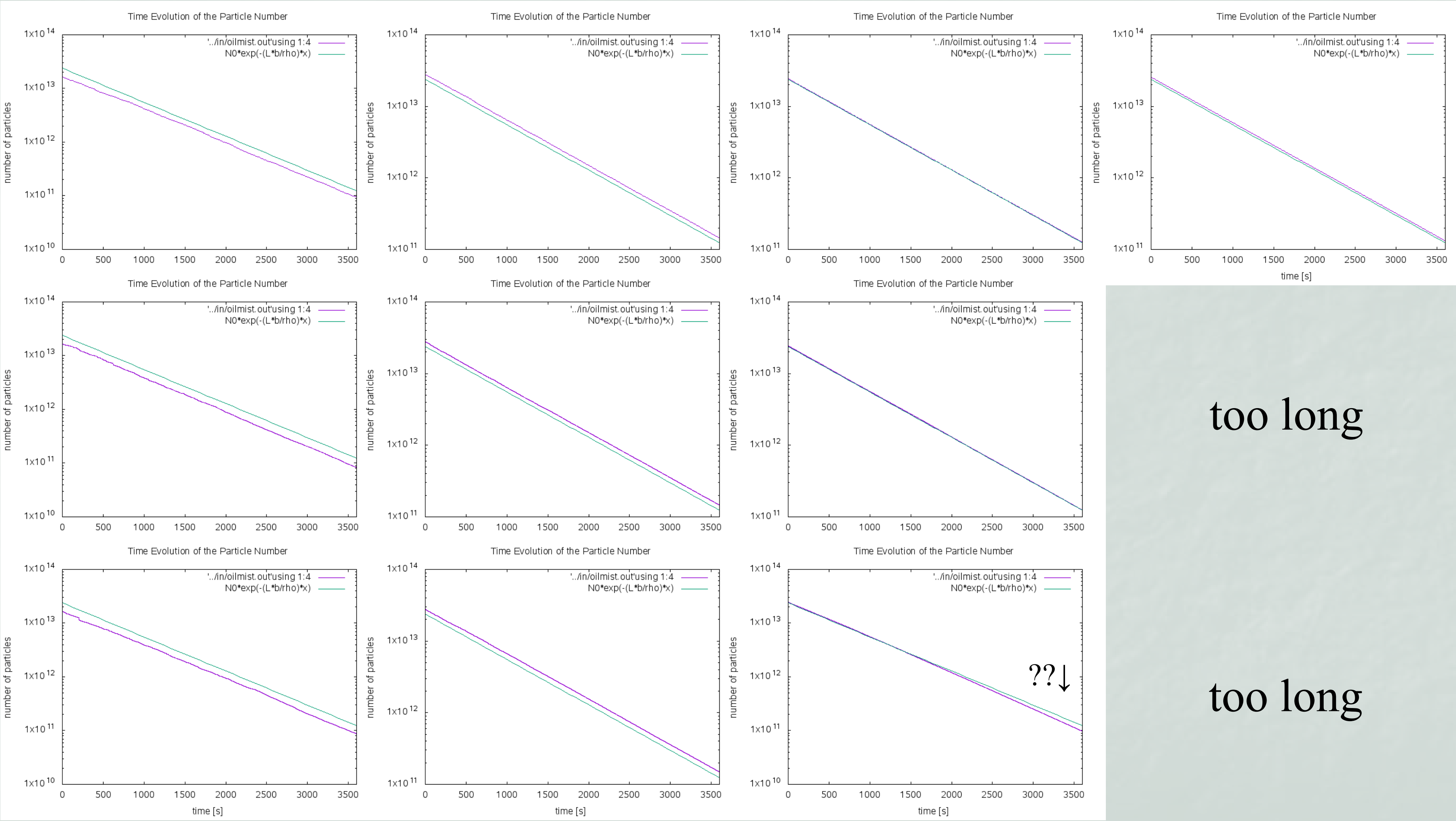
Time Evolution of droplet numbers (O'Rourke, Golovin)

N=80 | N=800 | N=8000 | N=80000

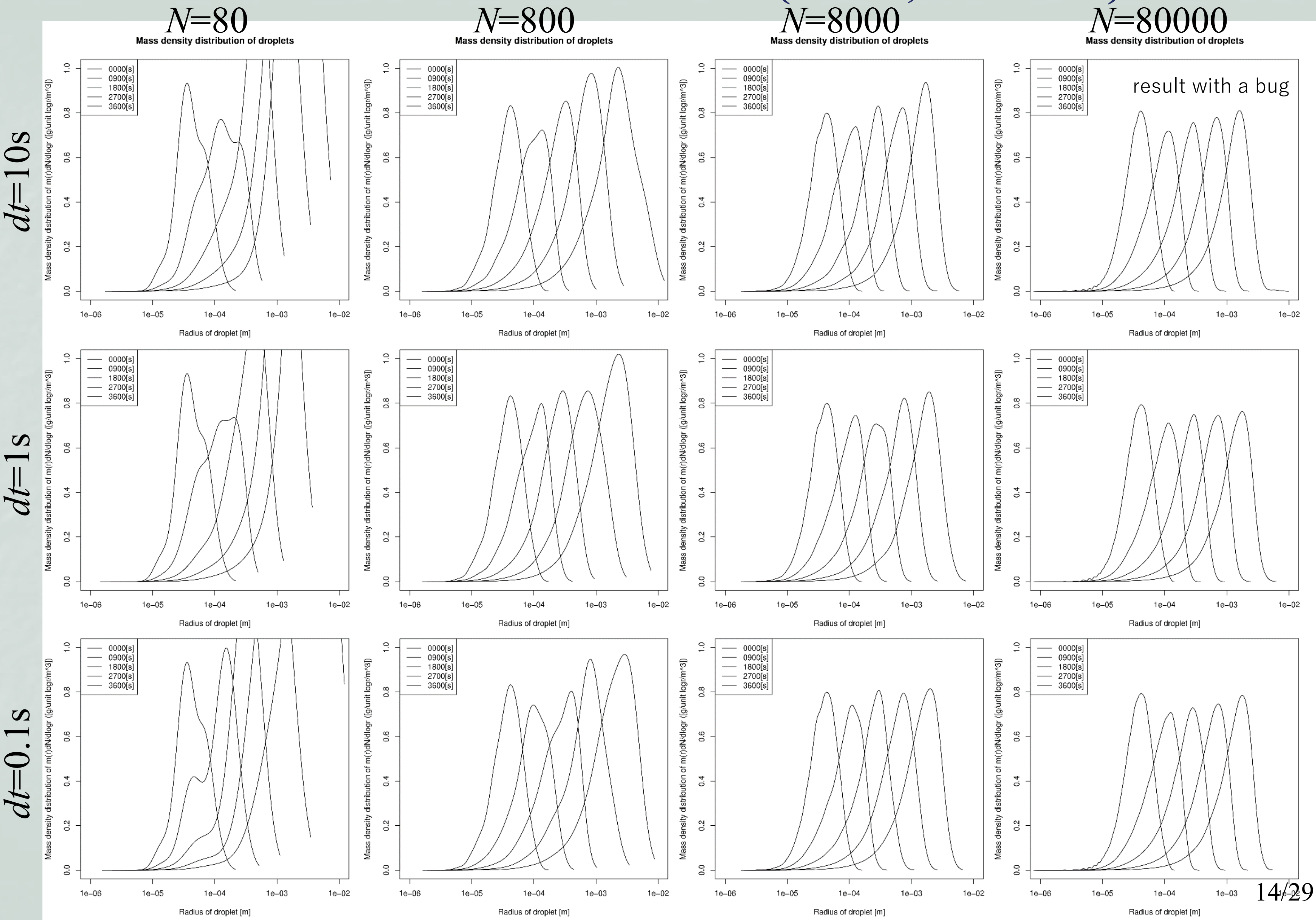
dt=10s

dt=1s

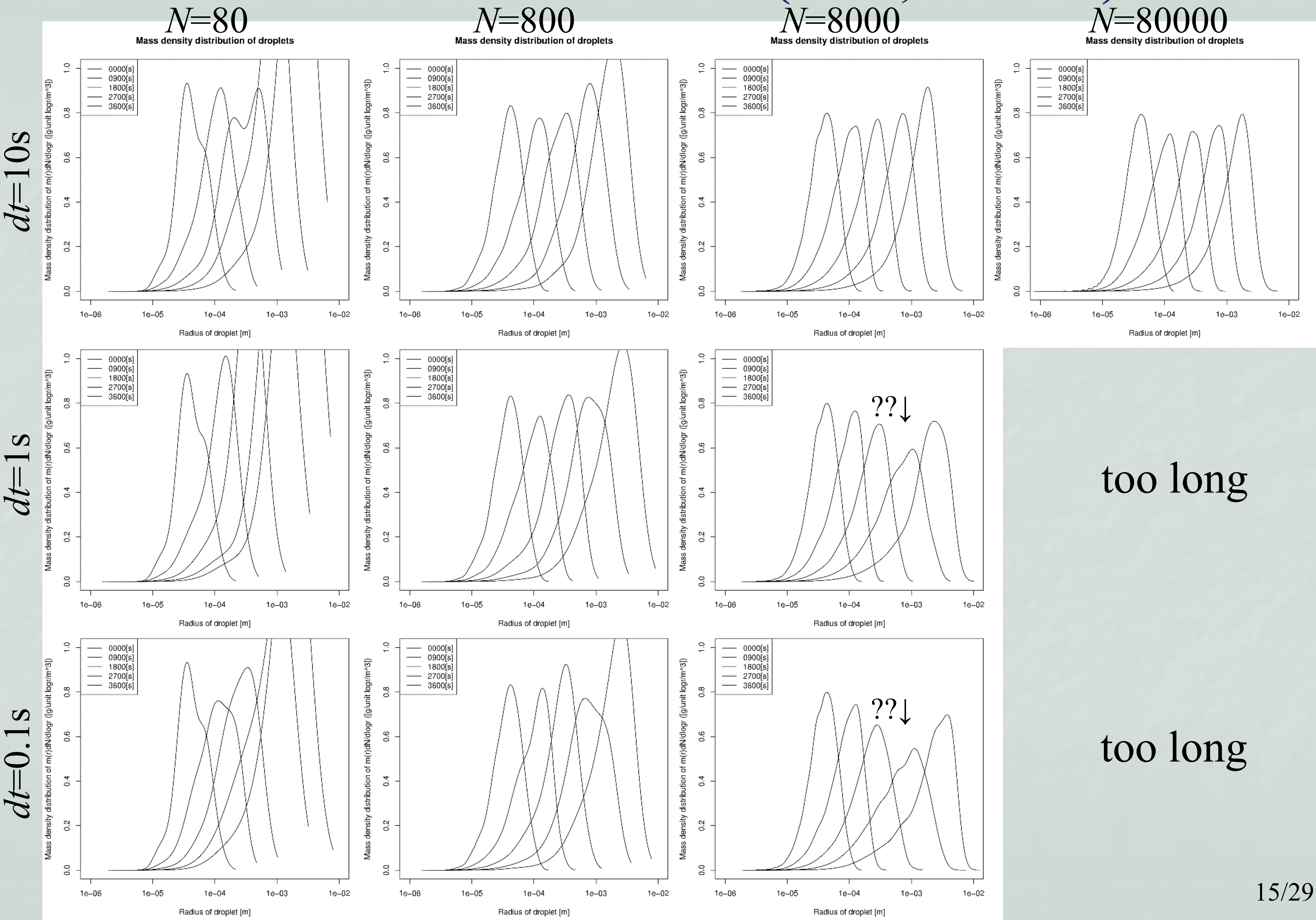
dt=0.1s



Time evolution of size distribution (SDM, Golovin)



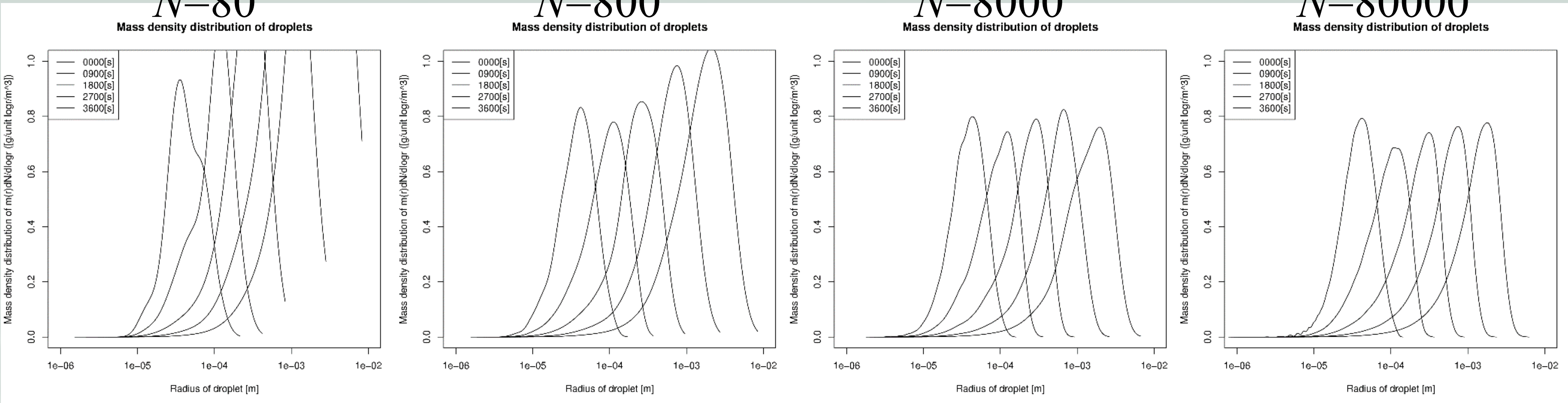
Time evolution of size distribution (NTC, Golovin)



Time evolution of size distribution (O’Rourke, Golovin)

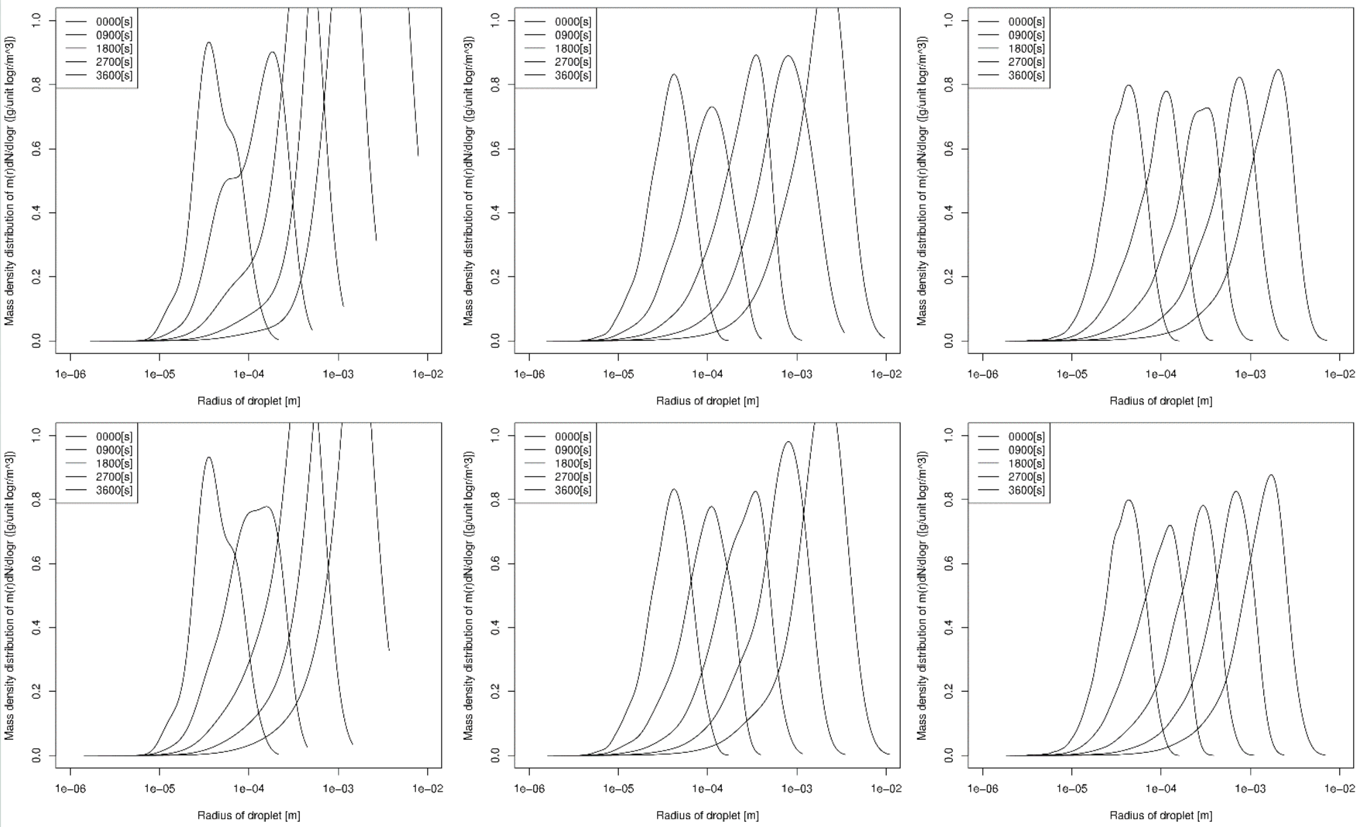
N=80 | N=800 | N=8000 | N=80000

dt=10s



dt=1s

dt=0.1s



too long

too long

Elapsed Time (Golovin)

SDM

Time [s]		parcel num											
		80			800			8000			80000		
		all	spray	IO	all	spray	IO	all	spray	IO	all	spray	IO
dt [s]	10	2	0	2	3	1	1	13	6	6	110	69	36
	1	21	0	8	28	9	10	77	35	24	884	546	291
	0.1	193	8	71	276	62	99	759	382	220	8508	5284	2770

NTC

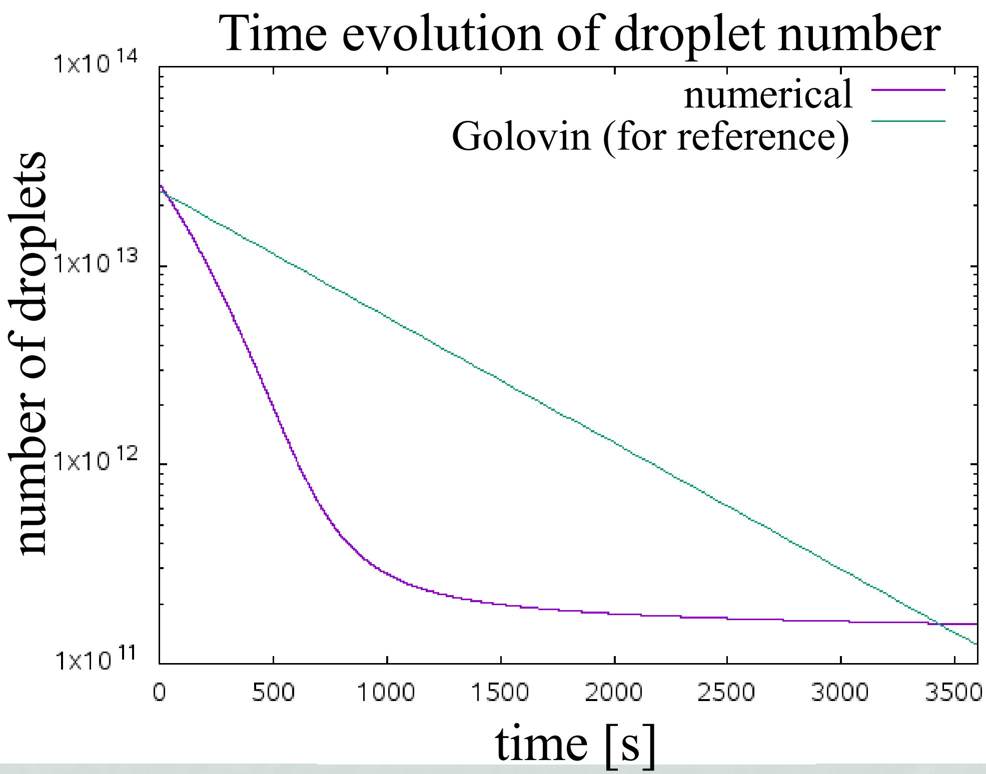
Time [s]		parcel num											
		80			800			8000			80000		
		all	spray	IO	all	spray	IO	all	spray	IO	all	spray	IO
dt [s]	10	4	0	1	11	9	0	669	668	1	143512	143474	31
	1	20	0	6	84	59	10	4529	4479	30		-	
	0.1	196	7	81	367	140	103	41384	41004	217		-	

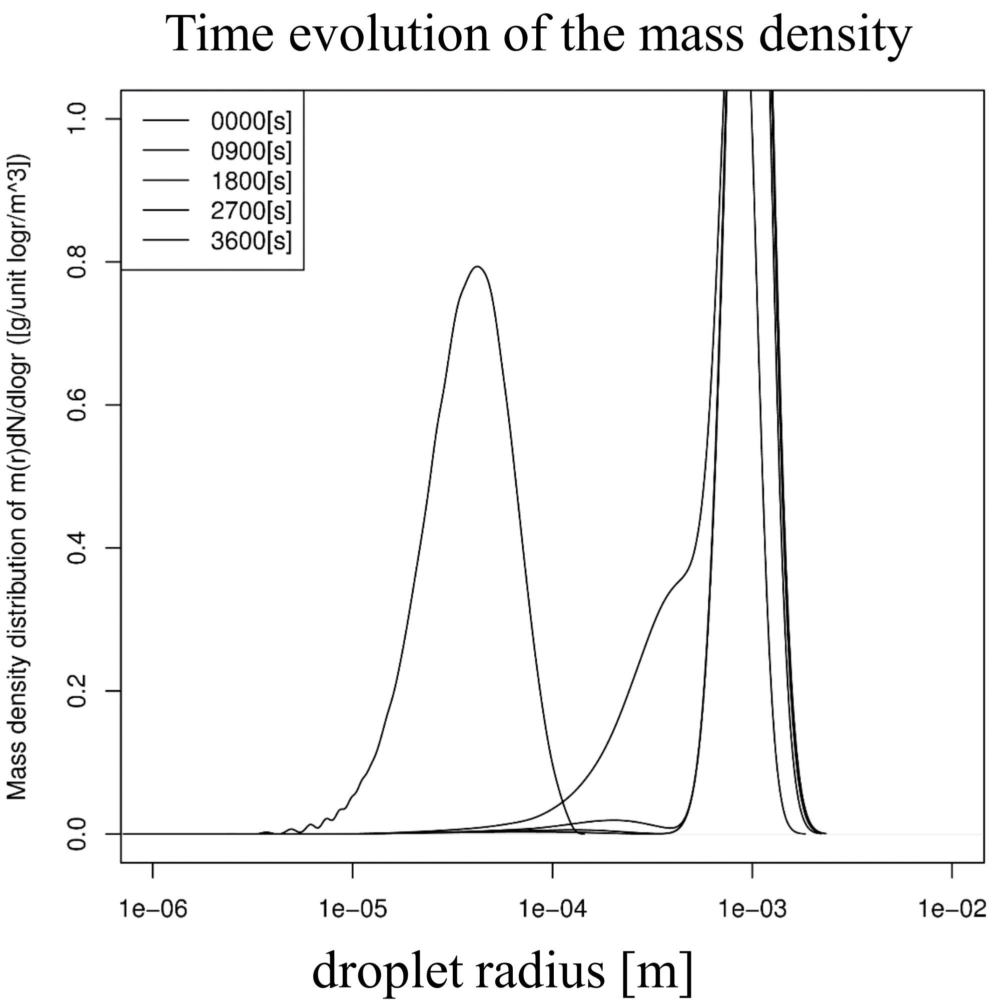
O'Rourke collision

Time [s]		parcel num											
		80			800			8000			80000		
		all	spray	IO	all	spray	IO	all	spray	IO	all	spray	IO
dt [s]	10	4	0	4	13	11	1	955	949	2	124283	124244	33
	1	23	1	12	126	104	8	9524	9472	30		-	
	0.1	207	19	63	1242	1025	87	94985	94538	242		-	

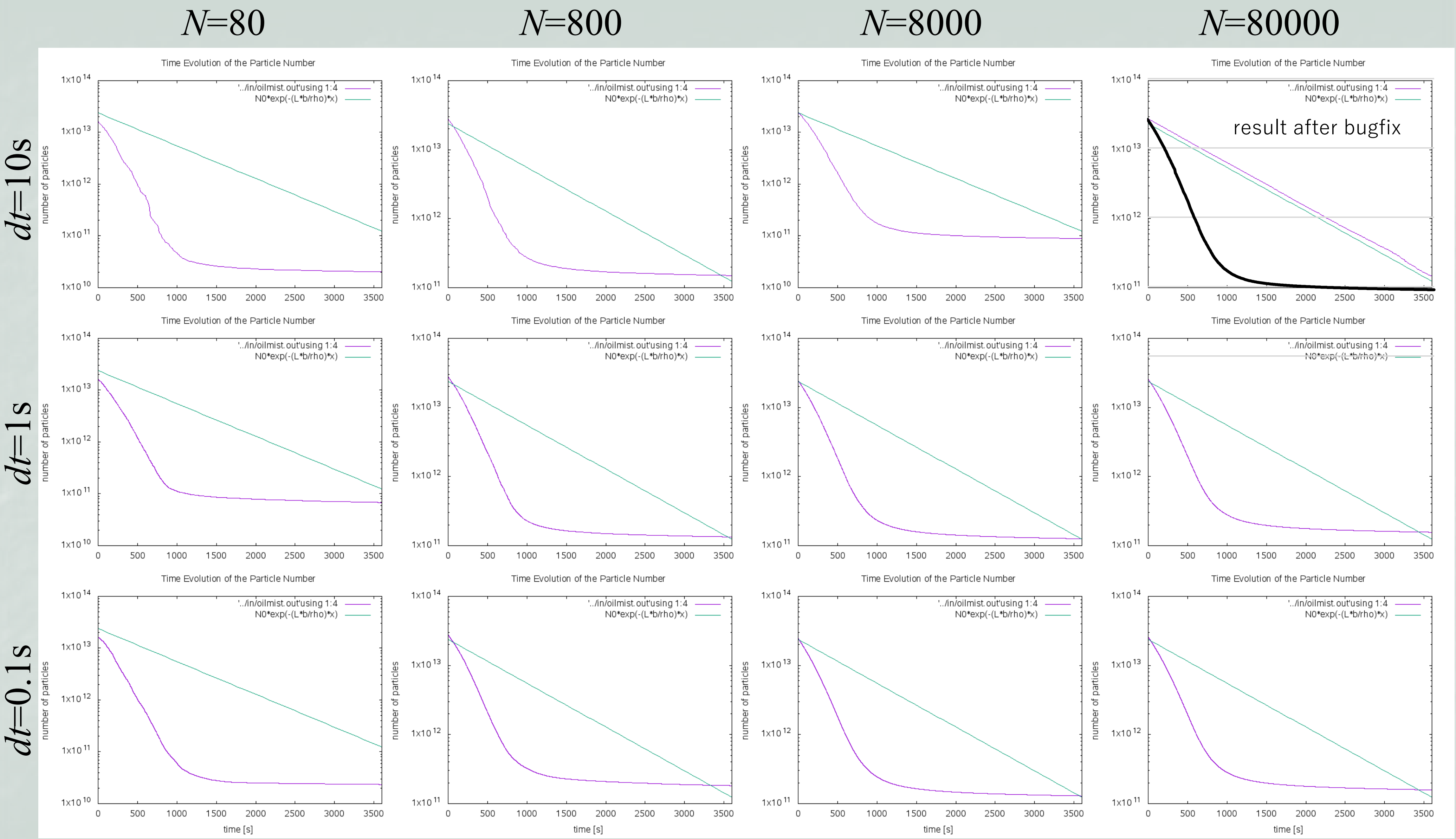
4. Result 2 (geometric kernel)

Correct behavior (SDM, N=80000, dt=0.1s)





Time Evolution of droplet numbers (SDM, geometric)



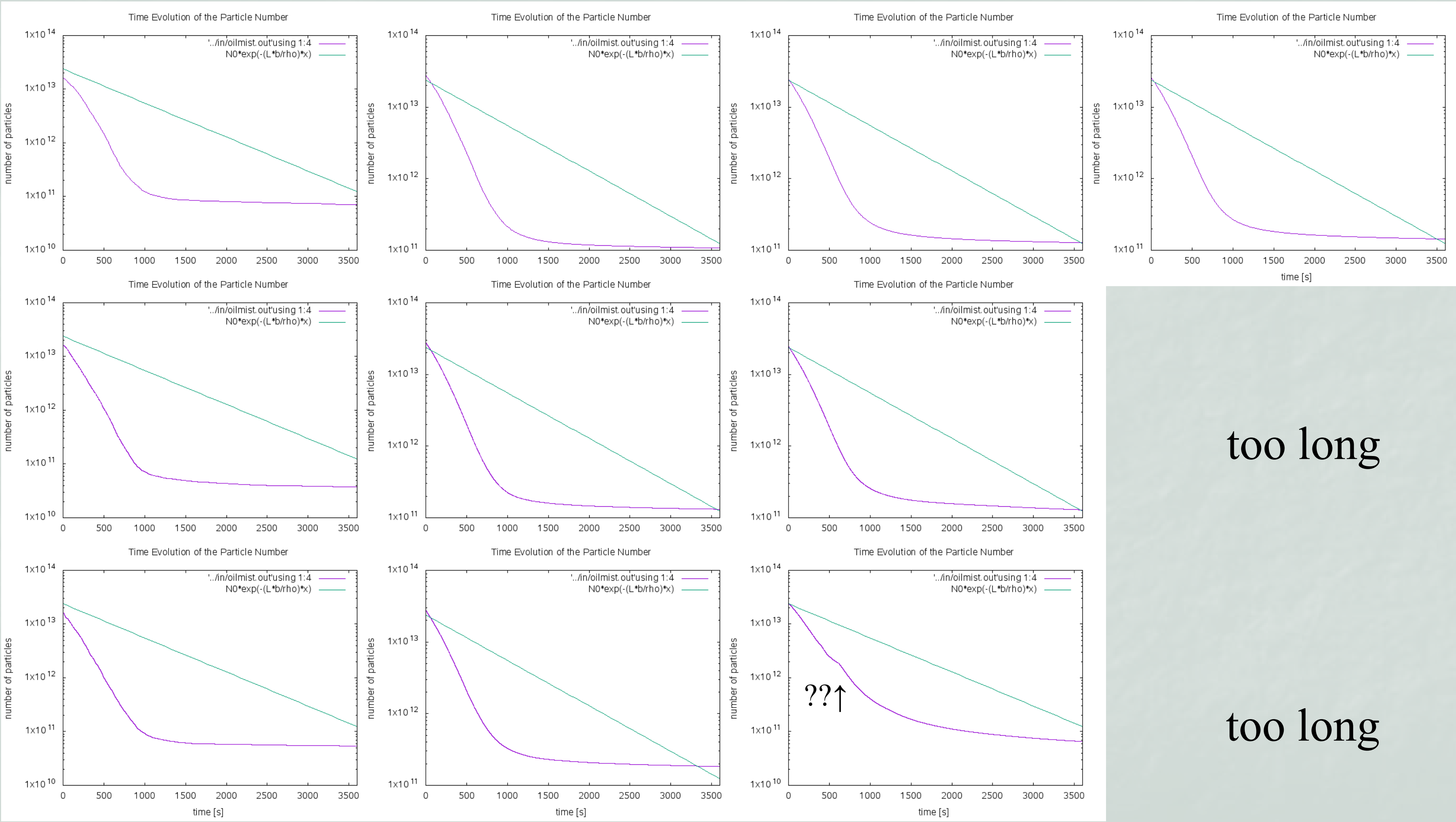
Time Evolution of droplet numbers (NTC, geometric)

N=80 N=800 N=8000 N=80000

dt=10s

dt=1s

dt=0.1s



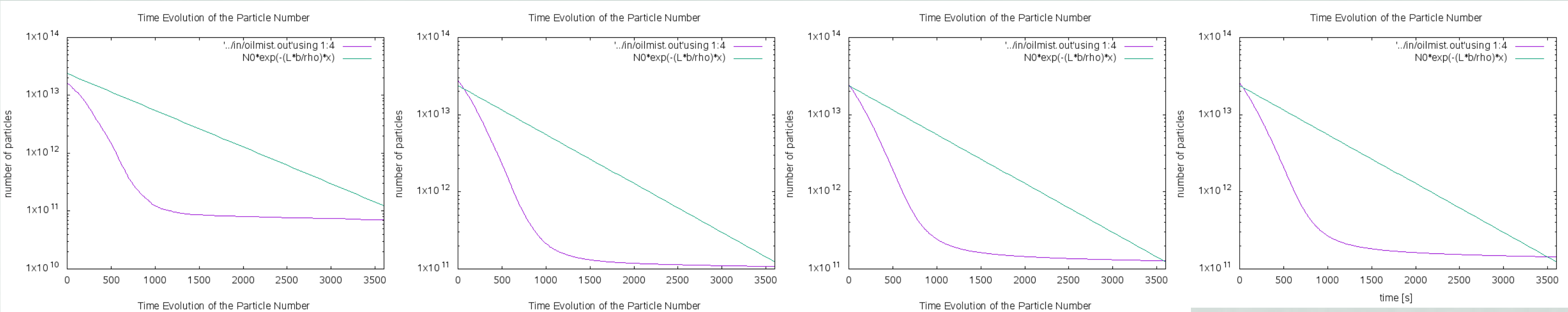
too long

too long

Time Evolution of droplet numbers (O'Rourke, geometric)

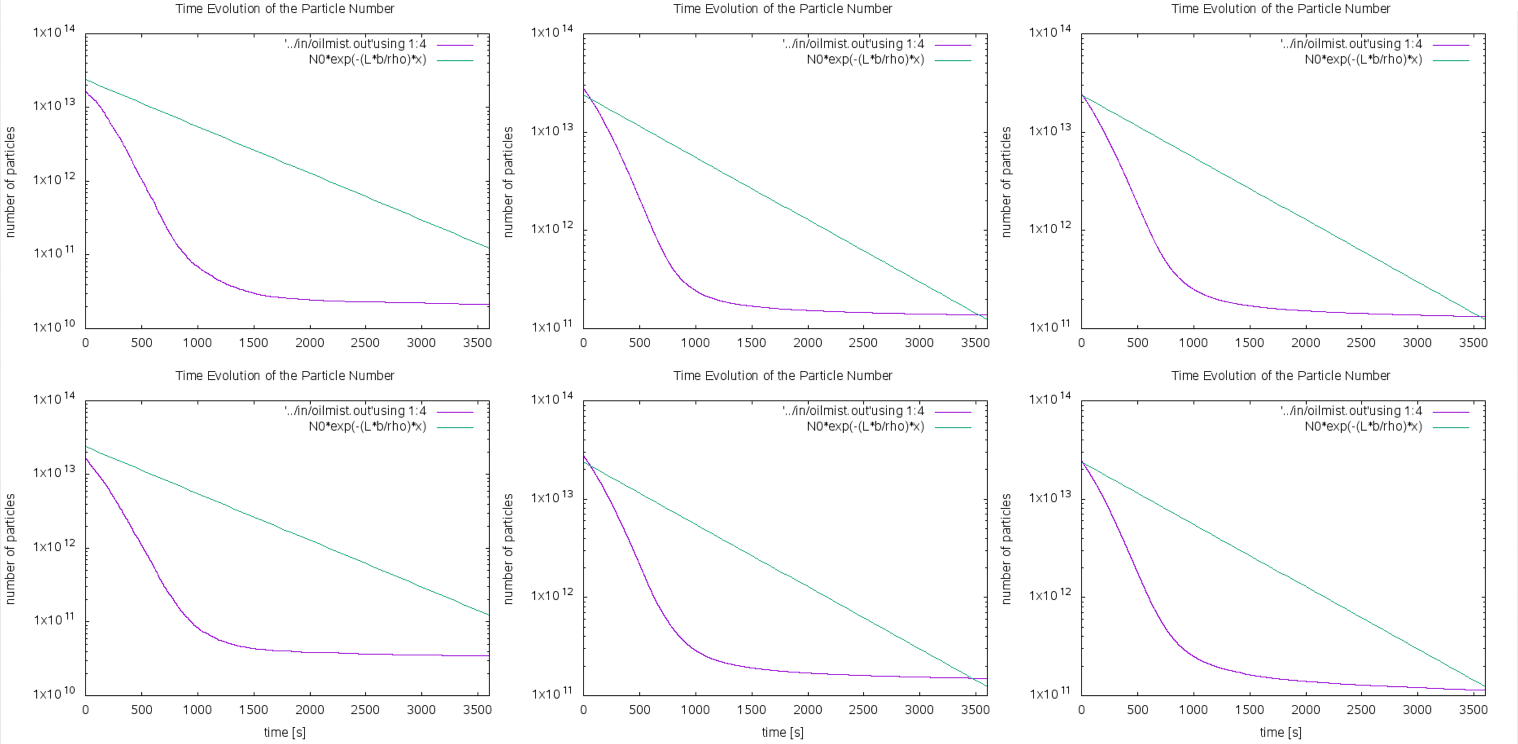
N=80 N=800 N=8000 N=80000

dt=10s



dt=1s

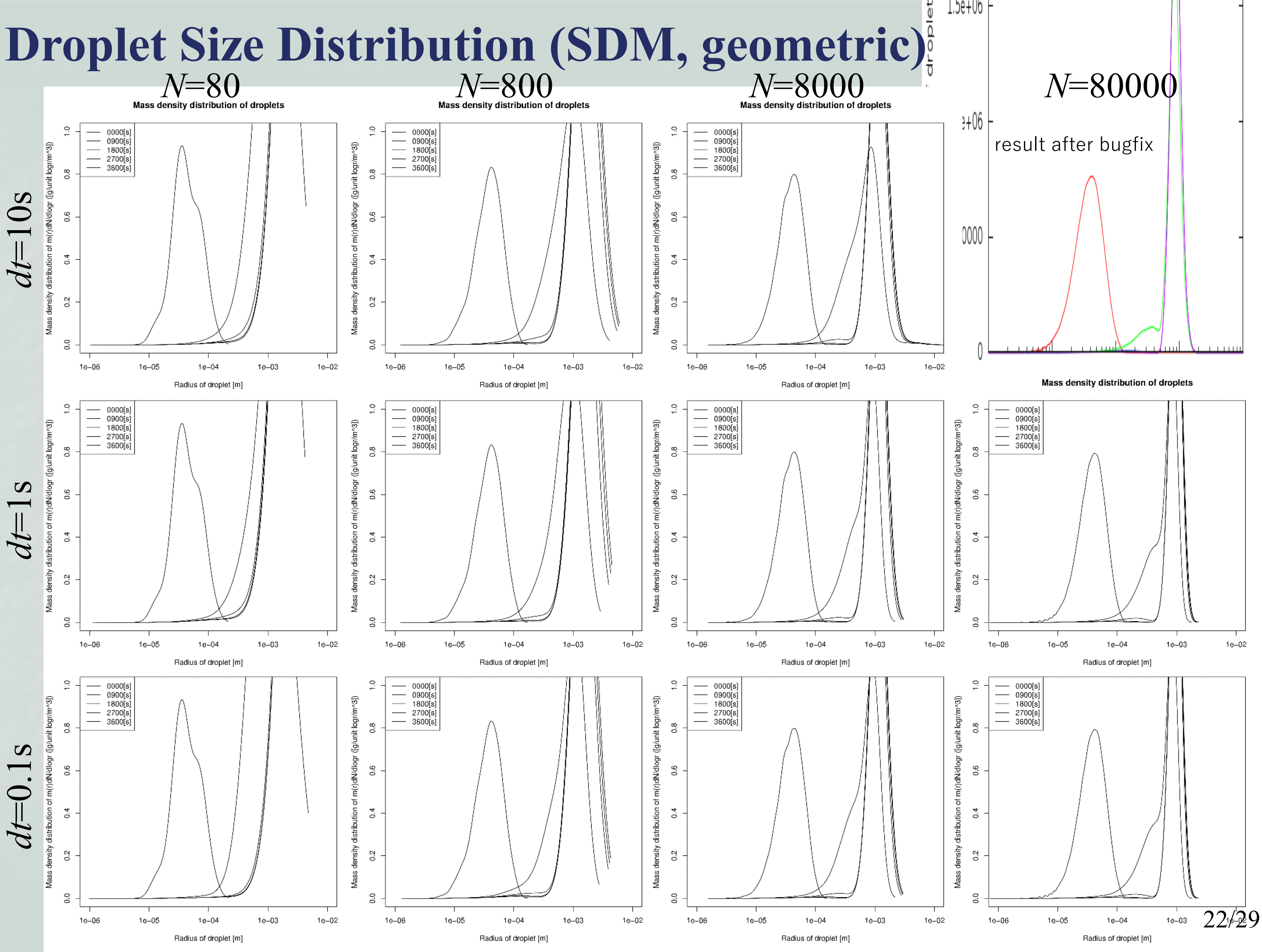
dt=0.1s



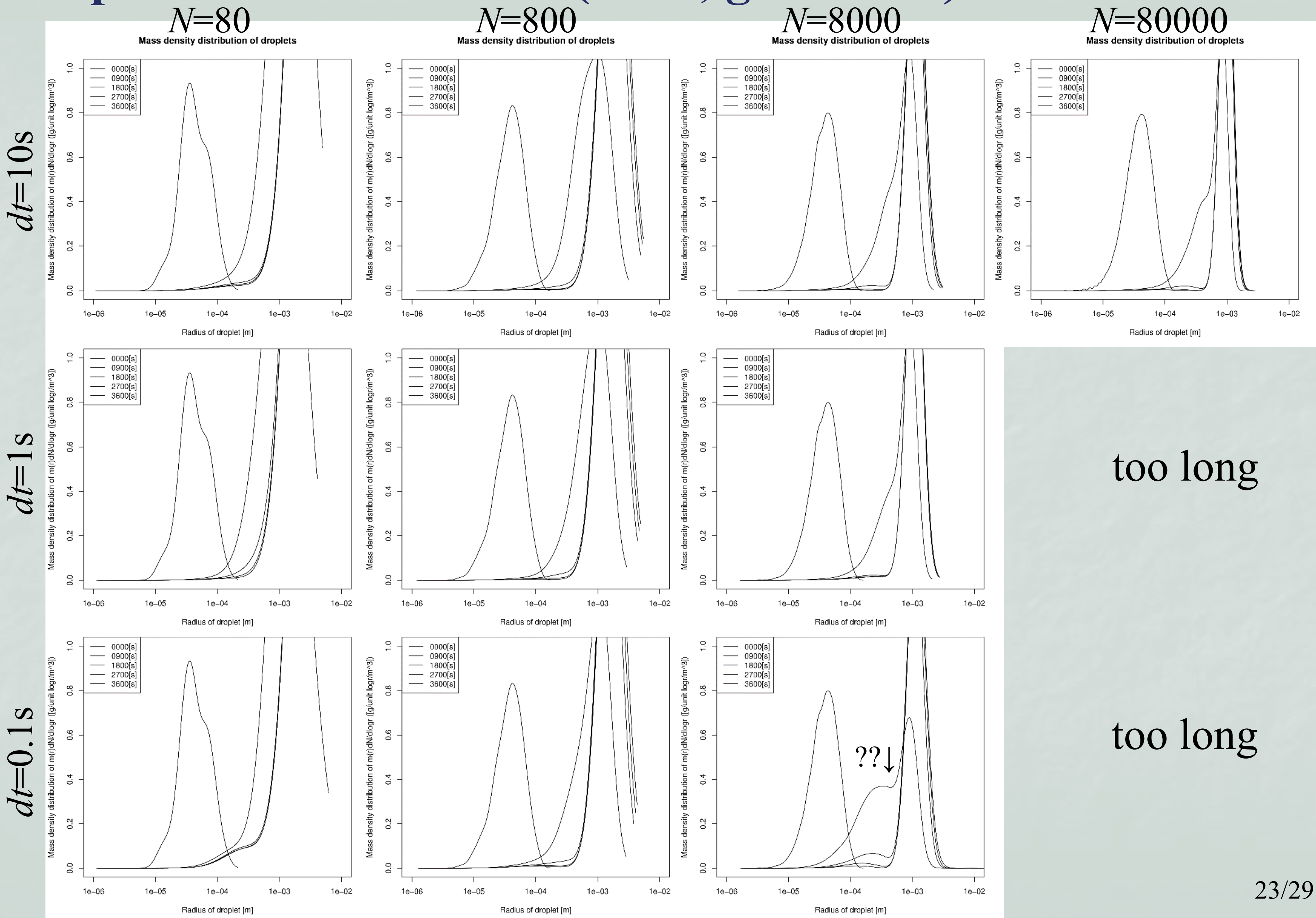
too long

too long

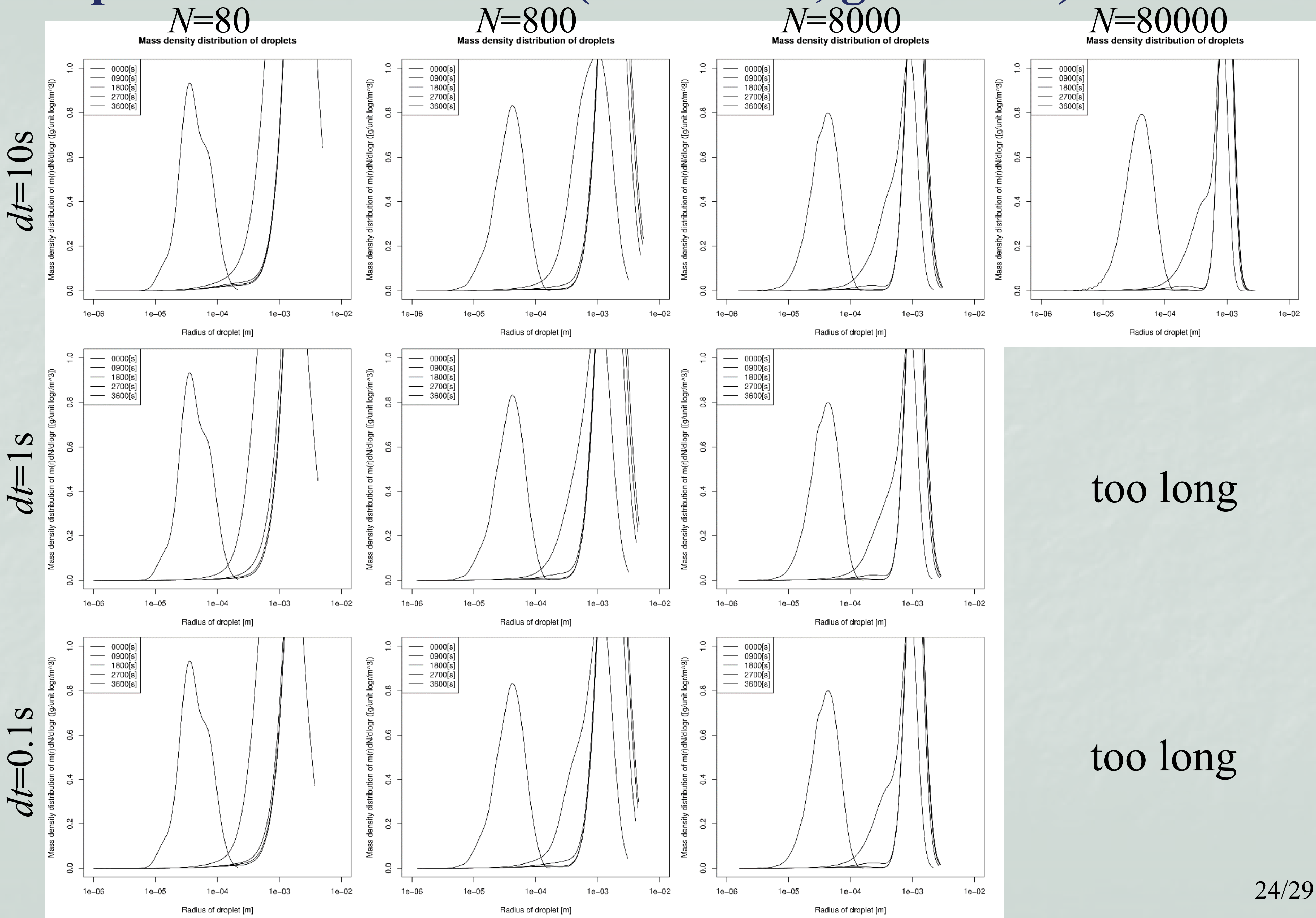
Droplet Size Distribution (SDM, geometric)



Droplet Size Distribution (NTC, geometric)



Droplet Size Distribution (O'Rourke, geometric)



Elapsed Time (geometric)

SDM

Time [s]		80			800			8000			80000		
		parcel num											
		all	spray	IO	all	spray	IO	all	spray	IO	all	spray	IO
dt [s]	10	2	0	0	3	0	2	10	4	6	106	69	34
	1	19	1	5	27	8	9	75	37	21	847	537	268
	0.1	188	2	67	269	56	82	721	338	233	8492	5331	2709

NTC

Time [s]		80			800			8000			80000		
		parcel num											
		all	spray	IO	all	spray	IO	all	spray	IO	all	spray	IO
dt [s]	10	3	1	1	14	8	5	1012	1009	1	139028	138987	33
	1	20	1	8	127	107	11	9210	9170	27		-	
	0.1	193	8	82	561	346	97	92660	92240	238		-	

O'Rourke collision

Time [s]		80			800			8000			80000		
		parcel num											
		all	spray	IO	all	spray	IO	all	spray	IO	all	spray	IO
dt [s]	10	3	0	1	13	10	1	1008	1002	4	161132	161093	32
	1	22	0	7	129	110	9	10107	10054	25		-	
	0.1	208	24	70	1276	1065	84	100639	100191	239		-	

6. Conclusion

Summary

Performance of three Monte Carlo schemes for collision-coalescence (SDM, NTC, and O'Rourke) were compared

It was confirmed that SDM outperforms the other two

- NTC is not O(N)? (Max prob. changes with SP num)

Future Work

There are some more Monte Carlo schemes

- DeVille et al. 2011 (Weighted Flow Algorithm): for aerosol dynamics. Implemented on PartMC
- Maybe some other schemes?

Note: Performance is very sensitive to how SPs are initialized

(see Unterstrasser et al., 2017)

A. Comment on Efficient Way of Initializing Super-Particles

How to initialize SPs is very flexible

Any SP population consistent with real particles can be used

- Constant multiplicity: Shima et al. (2009), Hoffmann et al. (2015)
- Grid: Unterstrasser et al. (2017), Dziekan and Pawlowska (2017)
- Uniform sampling: Arabas and Shima (2013), Sato et al., (2017, 2018)
- Constant mass: CONVERGE
- Quasi-random sampling: nobody tried yet?

Unterstrasser et al. (2017) concluded that for d=1 and D=0, "grid" (SingleSIP-init, multiSIP-init, and v_{random}-init) is far better than "constant multiplicity" (v_{const}-init)

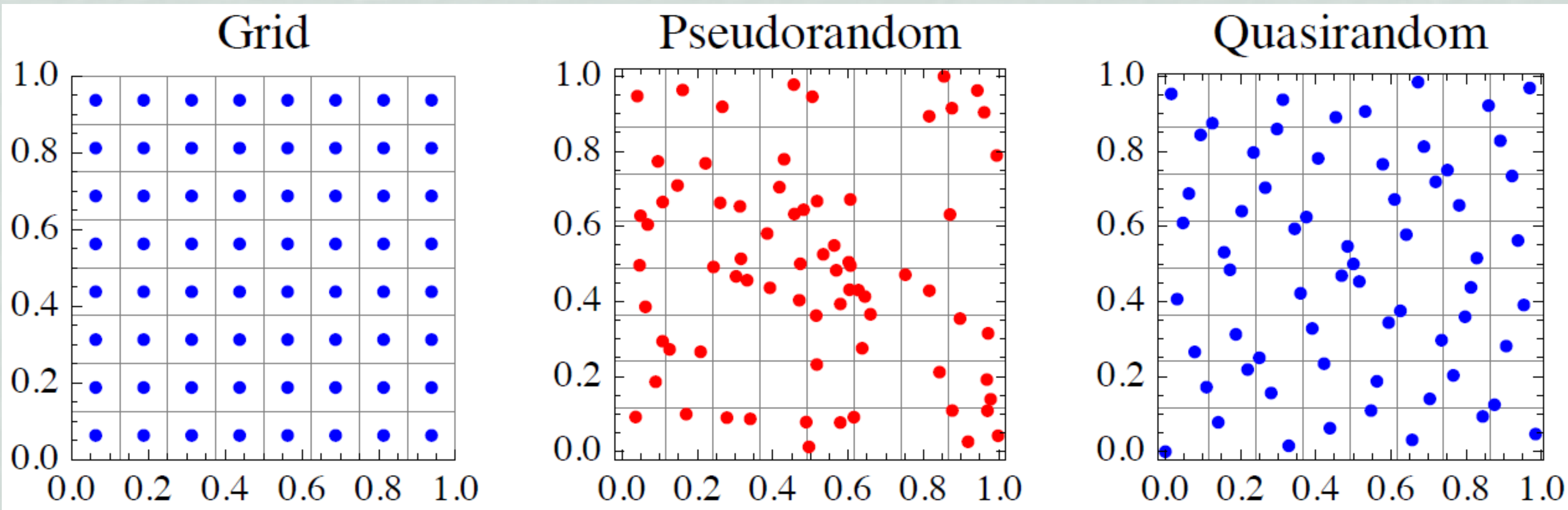
Discrepancy (e.g., Niederreiter, 1978)

Discrepancy of a set $P = \{x_1, ..., x_N\}$ is defined as

$$D_N(P) = \sup_{B \in J} \left| \frac{\#(B;P)}{N} - \lambda(B) \right|$$

In plain language, **"largest empty rectangular region that does not contain any points"**

from Carter (2011)



$D_N \propto N^{-1/s}$ $\quad D_N \propto N^{-1/2}$ $\quad D_N \propto (\log N)^s / N$

***s* is the dimension. Grid should not be used for *s*≧3**

Efficient Way of Initializing Super-Particles

In cloud models, $s = d$ (attribute number) + D (spatial dim)

I would argue, we should not use "gird" for initializing SPs, when $s \geqq 3$

"Uniform random sampling" should be better. But avoid using "constant multiplicity random sampling".

Perhaps we should try "quasi-random number"

It should be also a good idea to resample SPs adaptively. (Unterstrasser and Sölch (2014), Schwenkel et al. (2018))