NATIONAL SCIENCE CENTRE
POLAND

Erasmus+

UCDAVIS

AGH

Adaptive Time-Stepping within the Super-Droplet Method Monte-Carlo Coagulation

mini-workshop @Uni. Washington Seattle, Jan 2026

Emma Ware

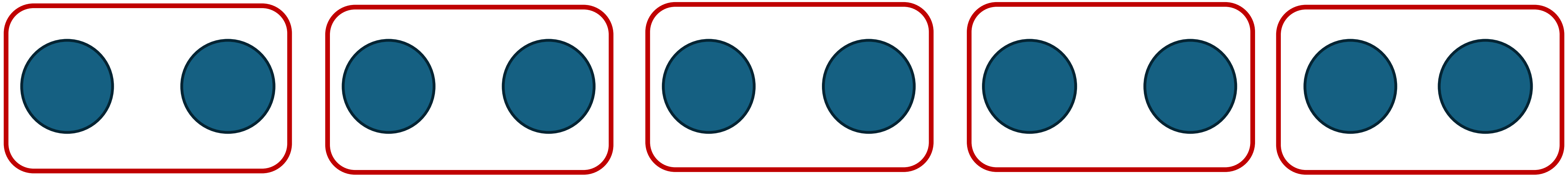
UCDAVIS

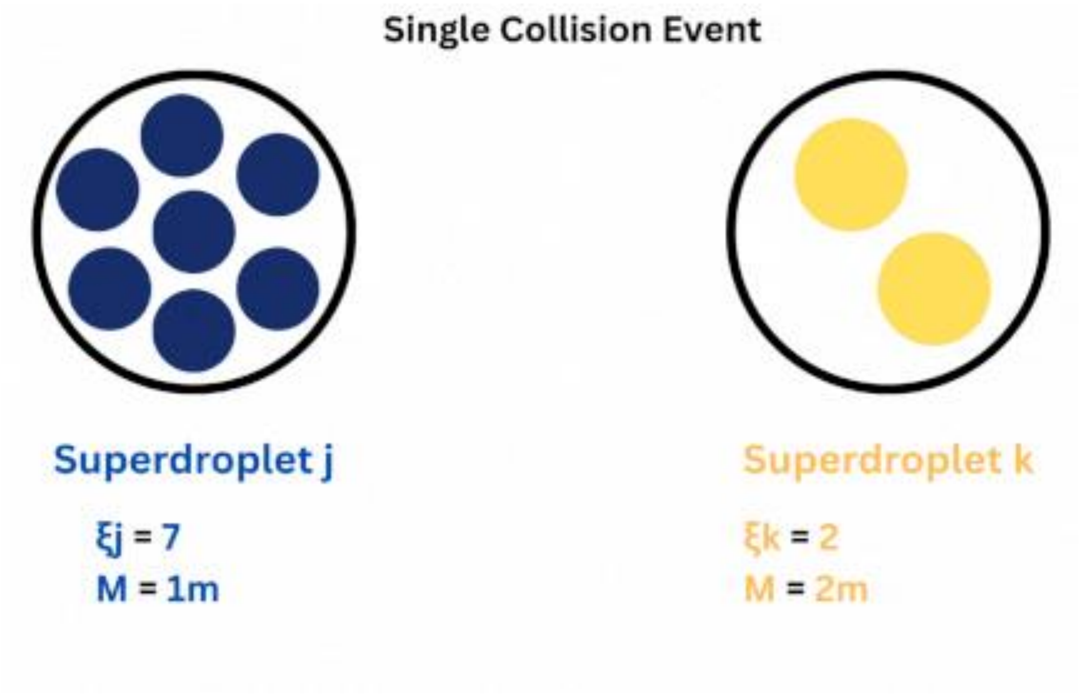
P.Bartman (Jagiellonian), S. Arabas (AGH), Adele Igel (UC Davis)

Super Droplet Method (Shima et al. 2009)

- When initialized efficiently, it is shown to outperform other algorithms in terms of convergence to analytic solutions (Unterstrasser et al, 2017)
- Monte-Carlo algorithm
- Tests pairs of superdroplets for collision events
- SDM Stochasticity resolves lucky rain formation (Morrison et al., 2024)

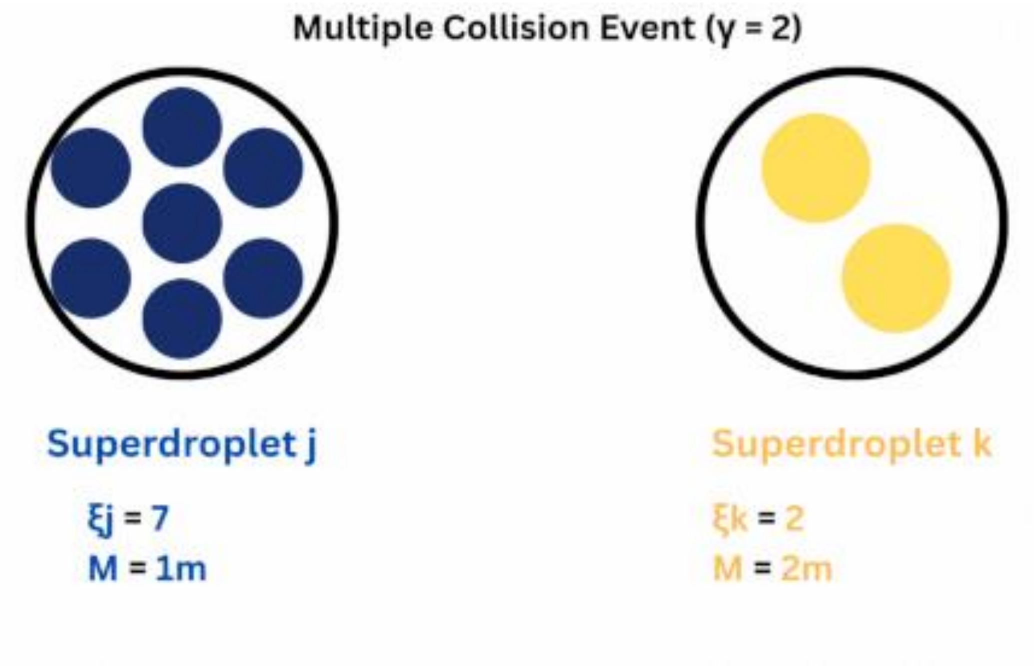
$$p_{pair\,j,k} = \frac{timestep}{collision\ volume} \cdot \xi_j \cdot \frac{total\ pairs}{pairs\ tested} \cdot Kernel(j,k), \qquad choosing\ \xi_j > \xi_k$$





- Choosing j and k so that $\xi_j > \xi_k$
- Superdroplet k collects ξ_k droplet from donator superdroplet j
- Superdroplet k grows in mass, superdroplet j decreases multiplicity

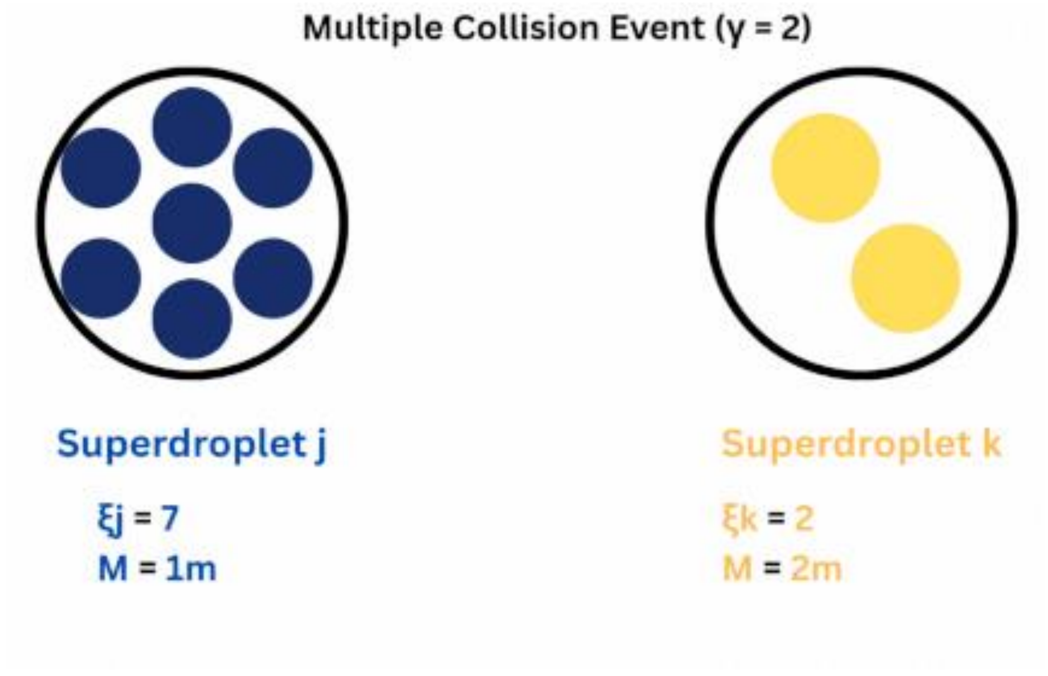
Mass and number of superdroplets are conserved



$\gamma =$ integer number of events
predicted by probability
and Monte-Carlo success

- Choosing j and k so that $\xi_j > \xi_k$
- Superdroplet k collects ξ_k droplet from donator superdroplet j
- Superdroplet k grows in mass, superdroplet j decreases multiplicity

Mass and number of superdroplets are conserved

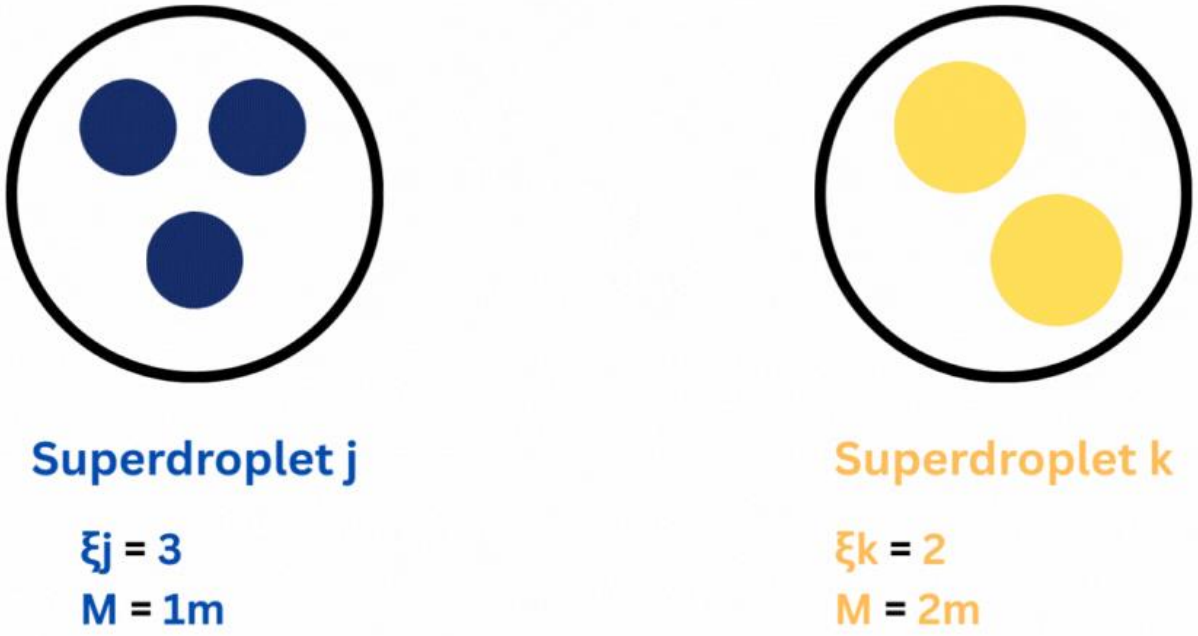


$\gamma =$ integer number of events predicted by probability and Monte-Carlo success

- Faster than bin methods in high dimensional attribute space
- Scales efficiently with number of attributes and particles
- Embarrassingly Parallel
- No numerical diffusion (retains particle identity)
- Conserves mass and Ns

- **Defining the Deficit Problem/Adaptive Timestepping**
- Adaptivity in a Box model
- Adaptivity in 2d prescribed flow

Deficit Problem (γ =2, ξj < γξk)



$$\gamma > \frac{\xi_j}{\xi_k}\,!!$$

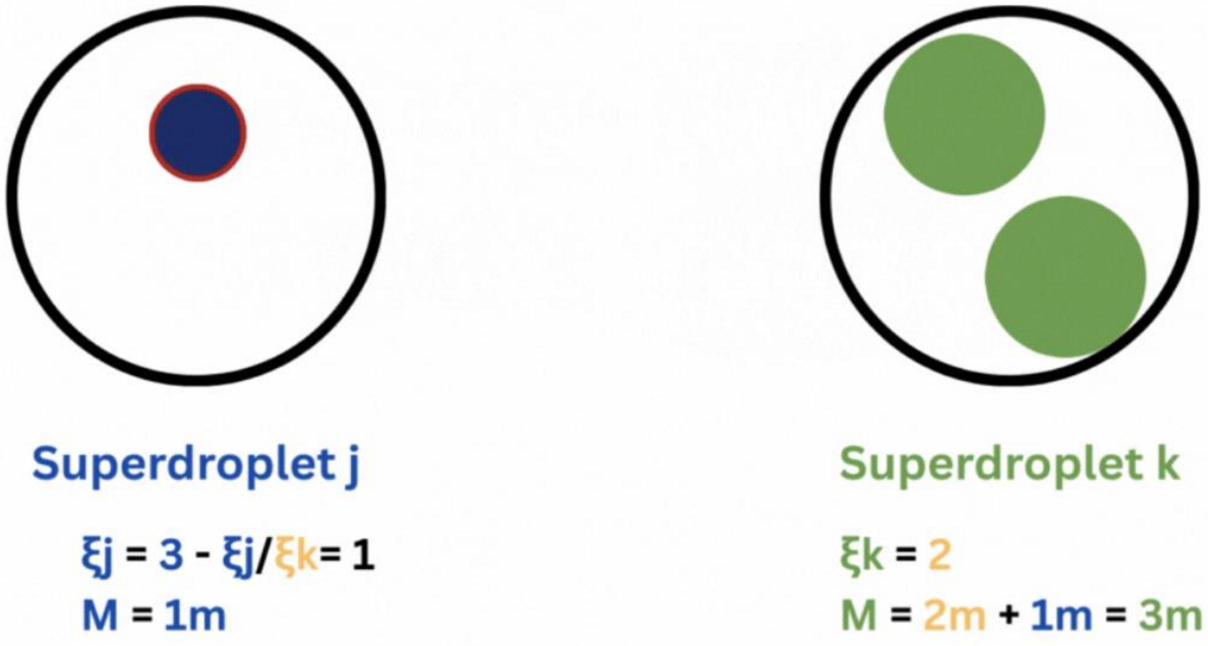
$$\downarrow$$

$$\gamma_{occured} = \left\lfloor \frac{\xi_j}{\xi_k} \right\rfloor$$

$$\gamma_{occured} < \gamma_{predicted}$$

- Shima et al., 2009 limit γ by $\frac{\xi_k}{\xi_j}$
- Pointed out in Dziekan and Pawlowska (2017)
- Quantified in Bartman, P., & Arabas, S. (2023).

Deficit Problem (γ =2, ξj < γξk)



$$\gamma > \frac{\xi_j}{\xi_k} !!$$

$$\gamma_{occured} = \left\lfloor \frac{\xi_j}{\xi_k} \right\rfloor$$

$$\gamma_{occured} < \gamma_{predicted}$$

- Shima et al., 2009 limit γ by $\frac{\xi_k}{\xi_j}$
- Pointed out in Dziekan and Pawlowska (2017)
- Quantified in Bartman, P., & Arabas, S. (2023).

$$deficit = (\gamma_{predicted} - \gamma_{occured}) \cdot \xi_k$$

Adaptive Time-Stepping Algorithm

$$deficit = (\gamma_{predicted} - \gamma_{occured}) \cdot \xi_k \qquad p_{pair\,j,k} = \frac{timestep}{collision\ volume} \cdot \xi_j \cdot \frac{total\ pairs}{pairs\ tested} \cdot Kernel(j,k),$$

We want to avoid $\gamma_{predicted} > \left\lfloor \frac{\xi_j}{\xi_k} \right\rfloor$:

Find the maximum timestep allowed for the pair:

$$p_{pair\,j,k} = \tilde{p}_{pair}\Delta t$$

$$\Delta t\_max_{pair} = \left\lfloor \frac{\xi_j}{\xi_k} \right\rfloor / \tilde{p}_{pair}$$

Find the limiting timestep for the cell:

$$\Delta t_{adaptive} = \min(\{\Delta t\_max_{pair} \ldots\}, \Delta t_{parent})$$

Repeat the substep until the model step is complete:

$$\Delta t_{left} = \Delta t_{parent} - \Delta t_{adaptive}$$

Adaptive Time-Stepping Analogies

If an algorithm does not have a way to handle multiple collisions, it must have an analogous method to constrain the probability < 1

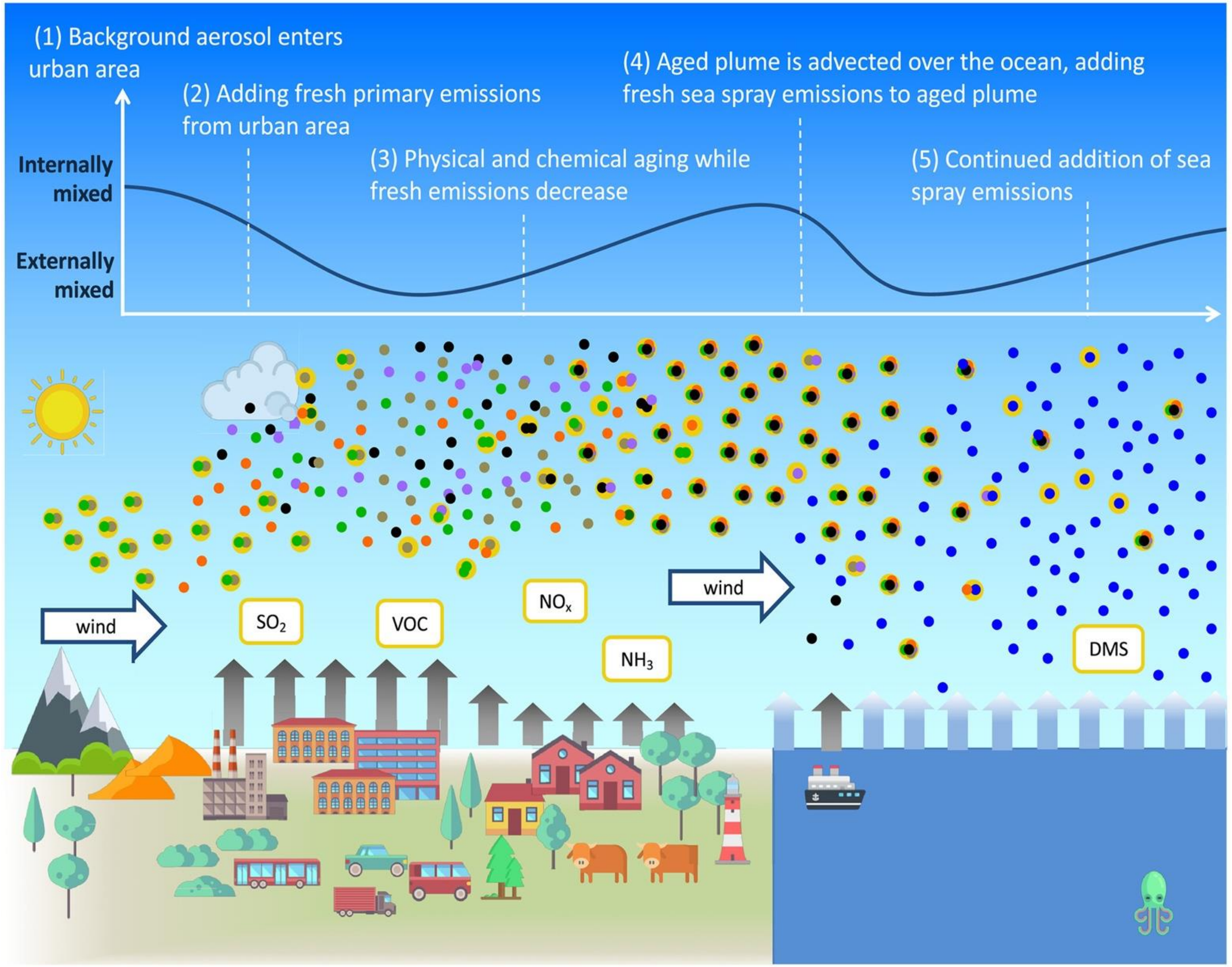
RemappingAlgorithm(RMA), Andrejczuk et al. (2010)

- RMA remaps between Superdroplets and binned mass every coalescence step
- Unterstrasser et al. (2017) proposes an adaptive substep when limited by bin concentration
- Comments on the stiffness of the Hall kernel when there are large droplets

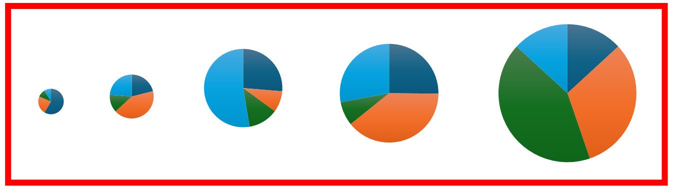
Weighted Flow Algorithm (WFA), Deville et al. (2011)

- used in PartMC (West, Riemer, et al., 2022)
- Particle-based Monte-Carlo algorithm
- Weighting is not a particle attribute
- optimizes the number of pairs tested out of the possible pairs

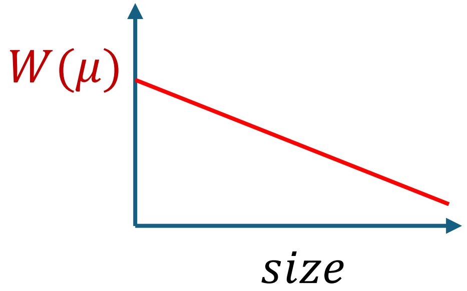
Adaptive Time-Stepping Analogies: WFA


(1) Background aerosol enters urban area
(2) Adding fresh primary emissions from urban area
(3) Physical and chemical aging while fresh emissions decrease
(4) Aged plume is advected over the ocean, adding fresh sea spray emissions to aged plume
(5) Continued addition of sea spray emissions
Internally mixed
Externally mixed
wind
SO_2
VOC
NO_x
NH_3
wind
DMS

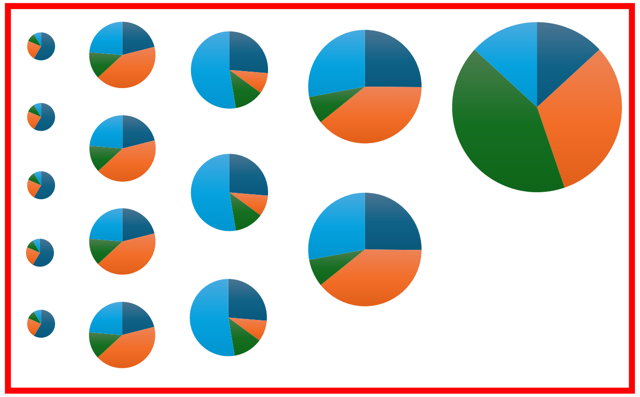
superparticle population:



weighting function:

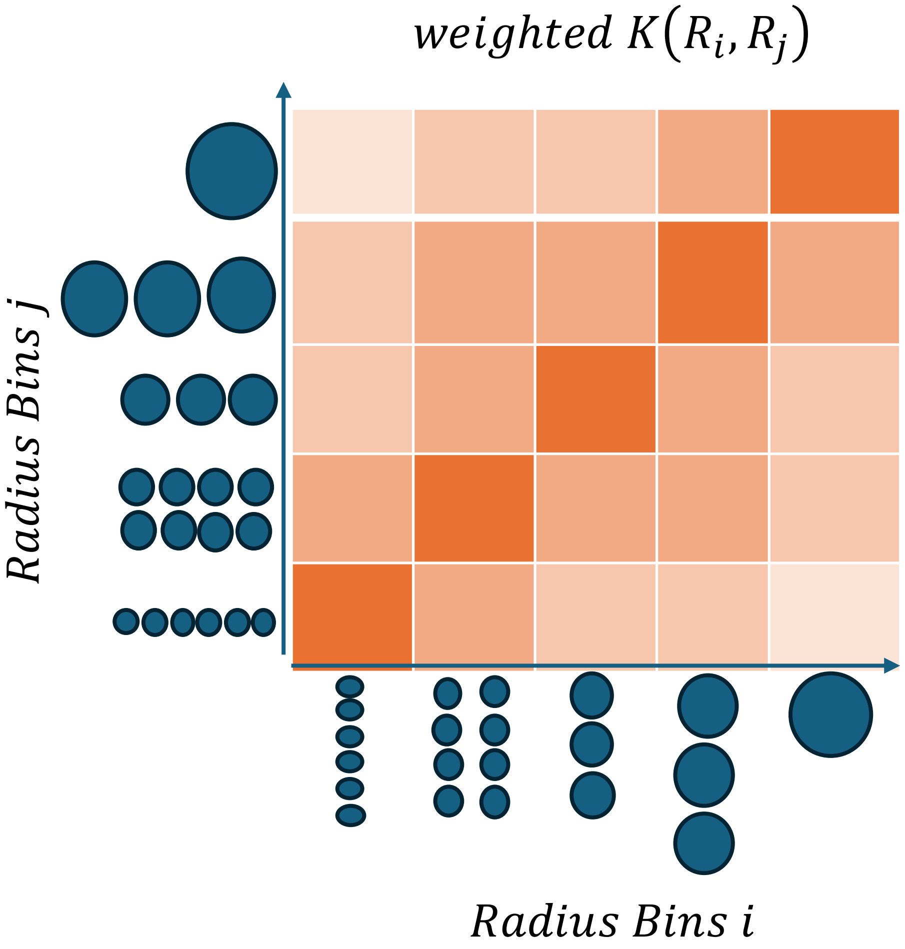


particle population:



Z. D'Aquino, et al.. PyPartMC (2024), N. Riemer, et al (2019), J. H. Curtis et al., (2017)

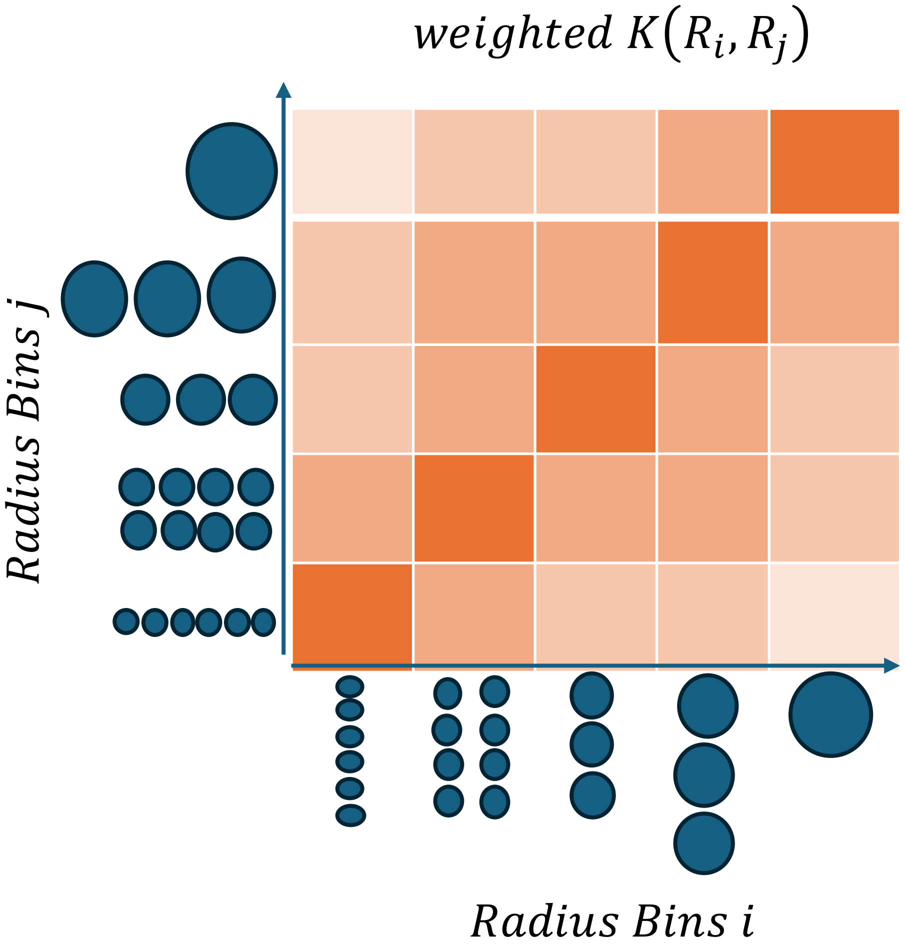
Adaptive Time-Stepping Analogies: WFA

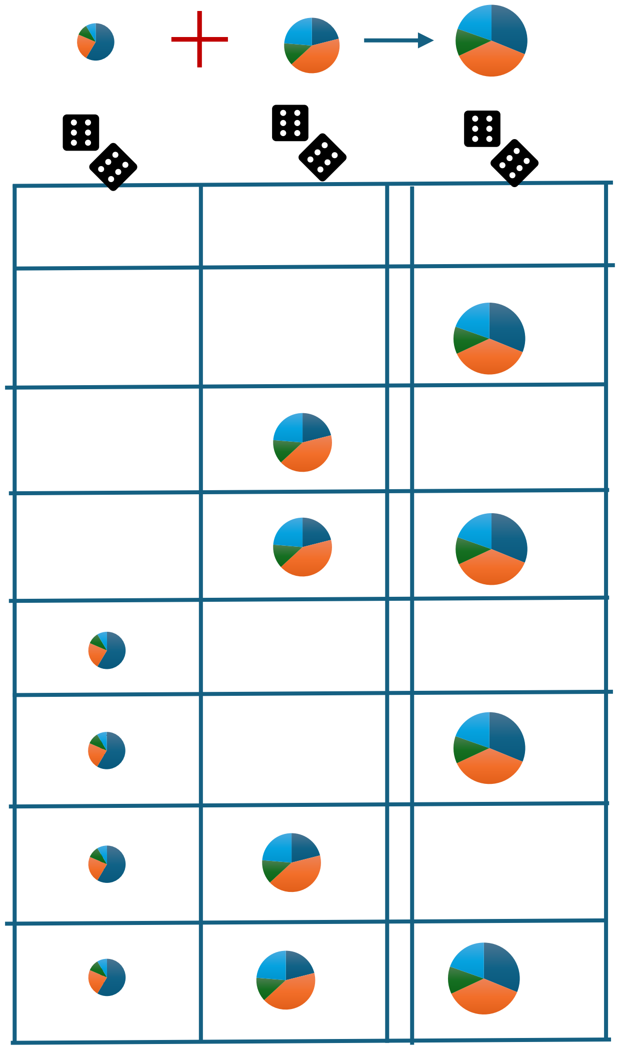


$$p_\alpha \propto \frac{total\ pairs\ in\ bin_{ij}}{\boxed{pairs\ tested}} \cdot weighted\ Kernel(R_i, R_j)$$

- Weighted Kernel of bin pair does not change in time
- Pairs tested is optimized at coarse bins by maximizing p_α to 1

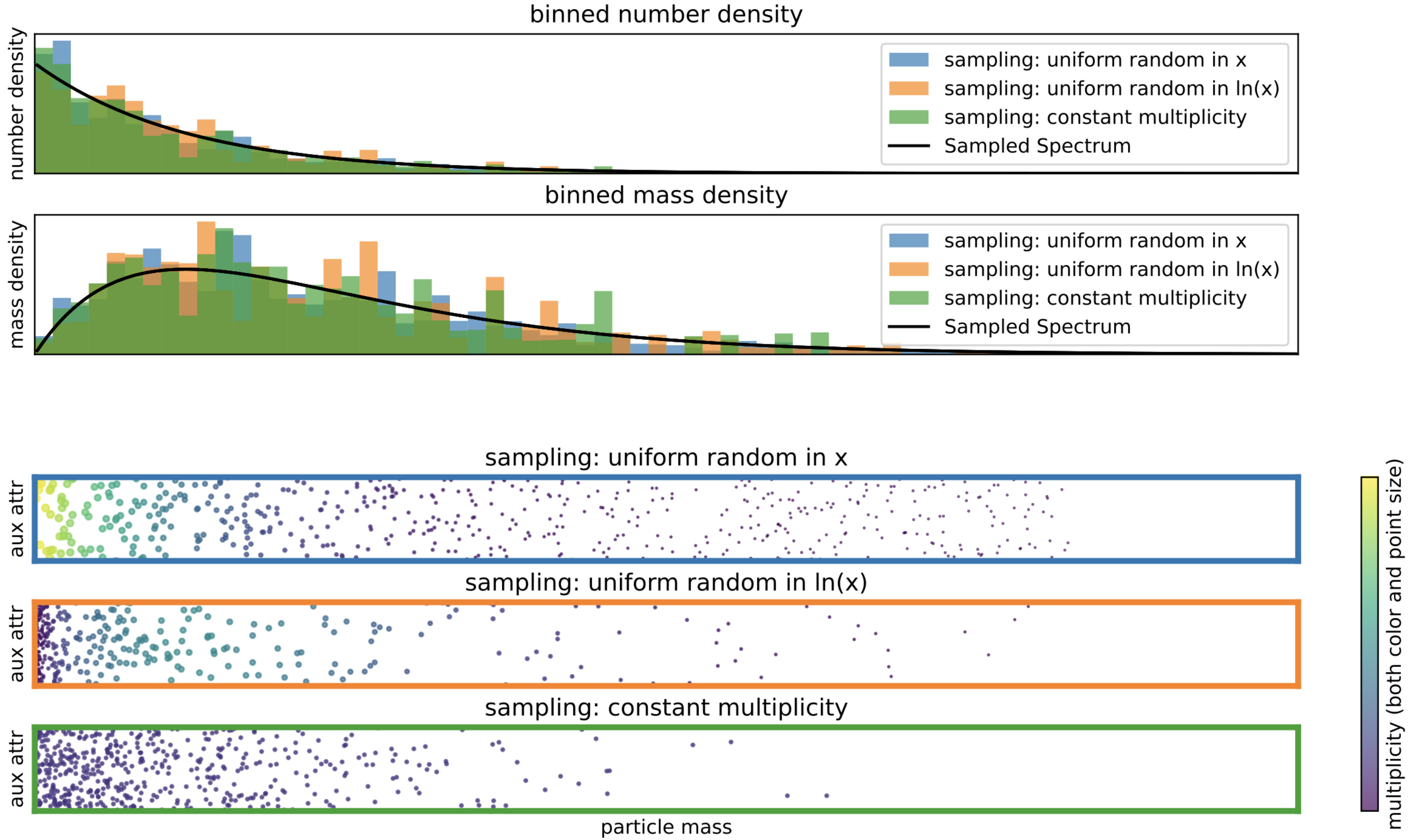
Adaptive Time-Stepping Analogies: WFA





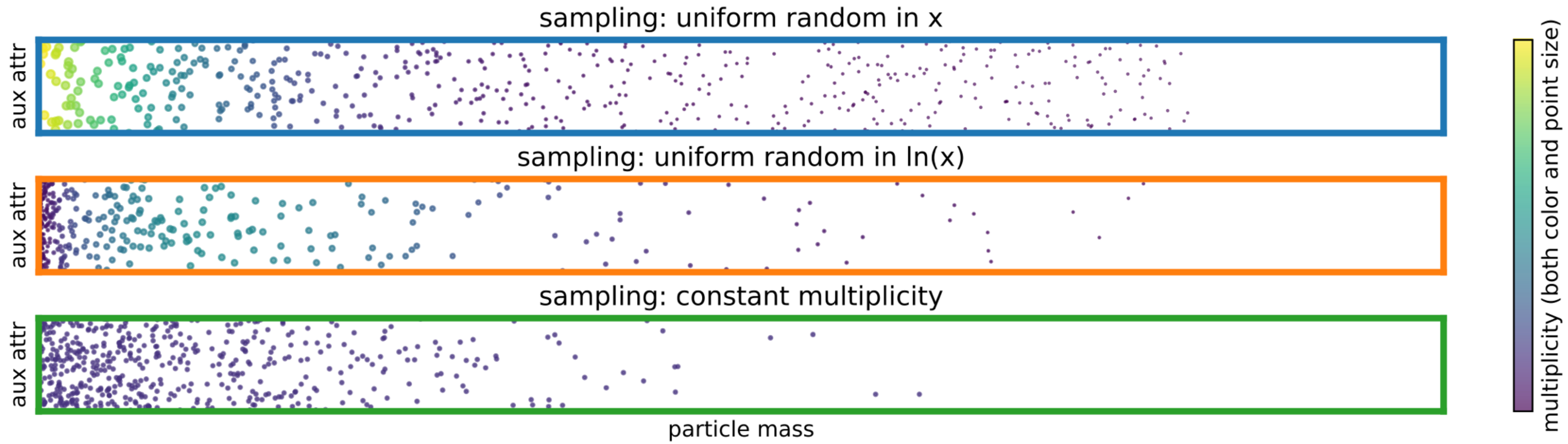
- Weights are helpful for the optimization
- Resulting superparticle might not be the weight of adding two smaller superparticles
- All 3 are probability checked for existence after a collision event based on their weights

Initialization



Initialization

- Optimization and comparison of initialization is explored in Unterstrasser et al. 2017 and 2019, Dziekan and Pawlowska 2017, Matsushima et al. 2023)
- Initializations with dynamic ranges of multiplicities are shown to do best
- Matsushima et al. (2023) proposed a new init method that parameterizes this dynamic spread



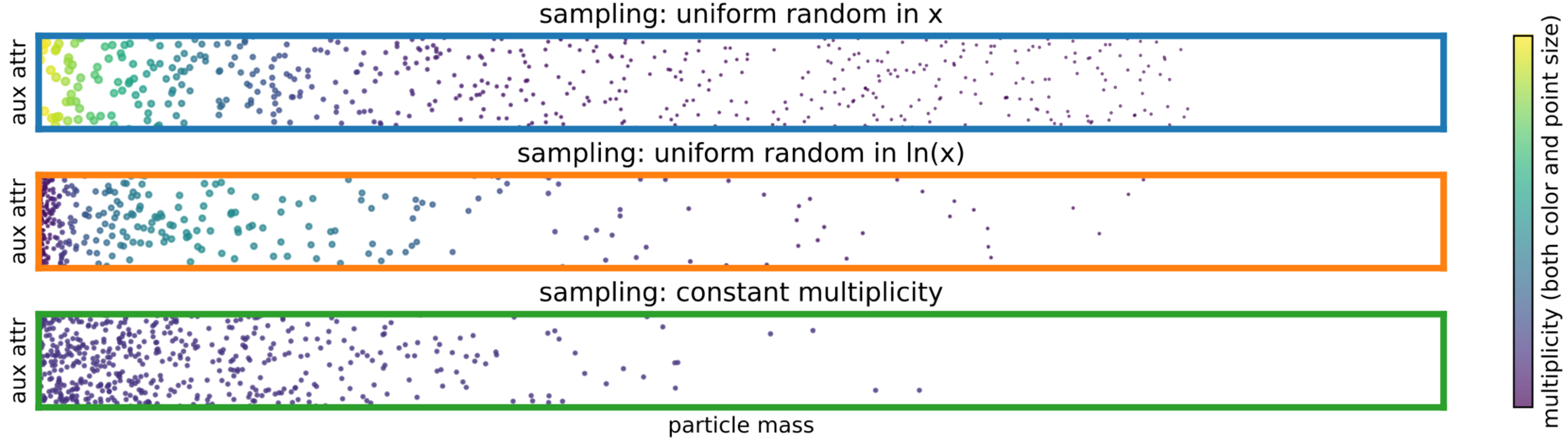
Initialization

$$p_{superdroplet\ pair} = \xi_j \cdot p_{droplet\ pair},$$ ⇧ choose $\xi_j > \xi_k$

$$droplet\ collisions\ per\ event = \xi_j \cdot \xi_k \cdot p_{droplet\ pair}$$ ⇩

Dynamic range

It is advantageous to have many superdroplet events of small multiplicities ξ_k



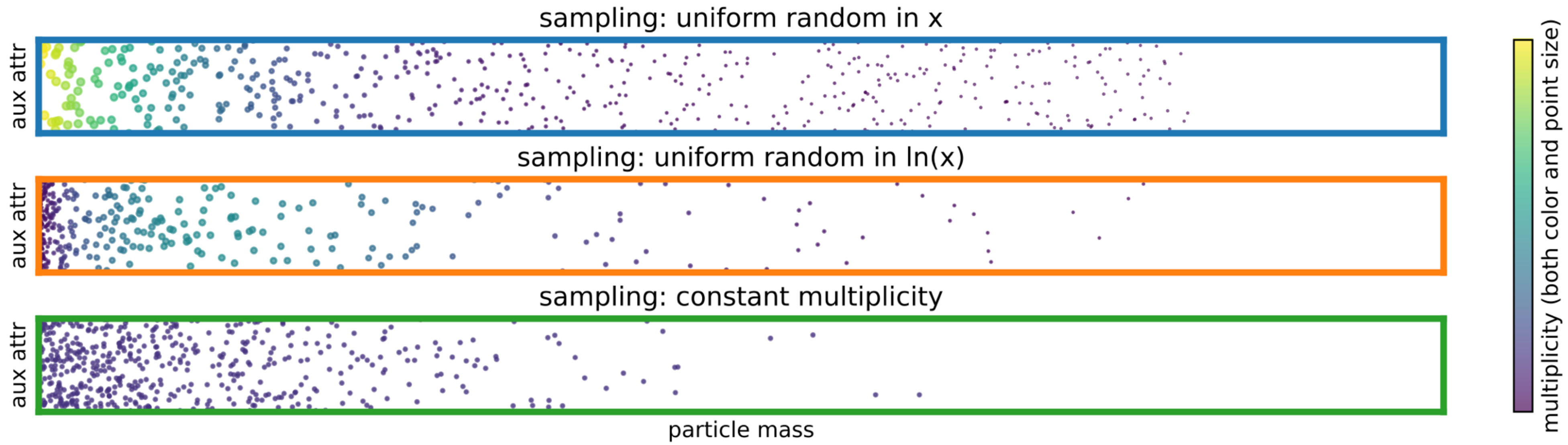
Initialization

$$p_{superdroplet\ pair} = \xi_j \cdot p_{droplet\ pair}, \Uparrow \qquad choose\ \xi_j > \xi_k$$

$$droplet\ collisions\ per\ event = \xi_j \cdot \xi_k \cdot p_{droplet\ pair} \Downarrow$$

Dynamic range

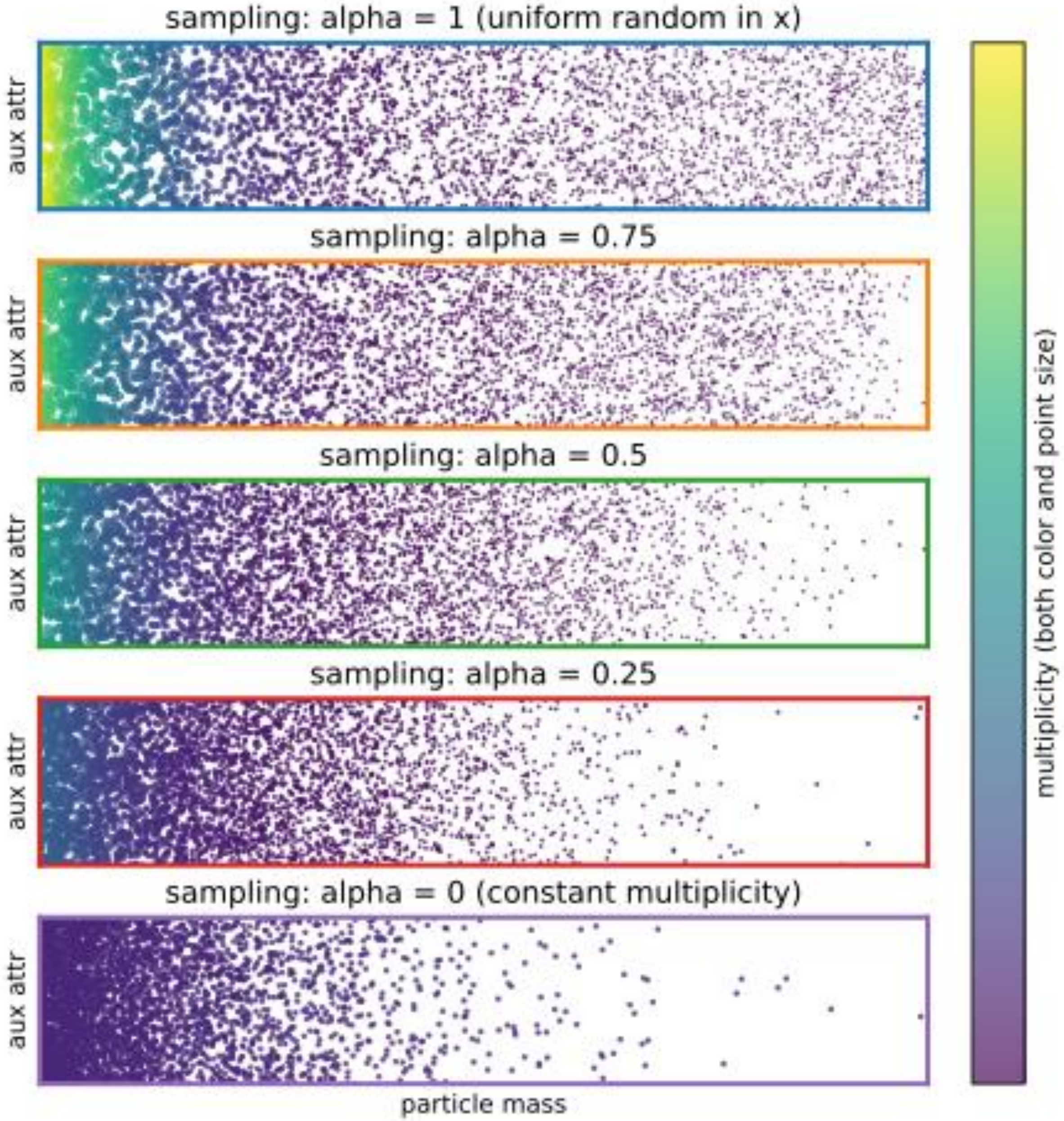
Lucky for deficit, dynamic range also maximizes $\overline{\xi_j/\xi_k}$



Initialization

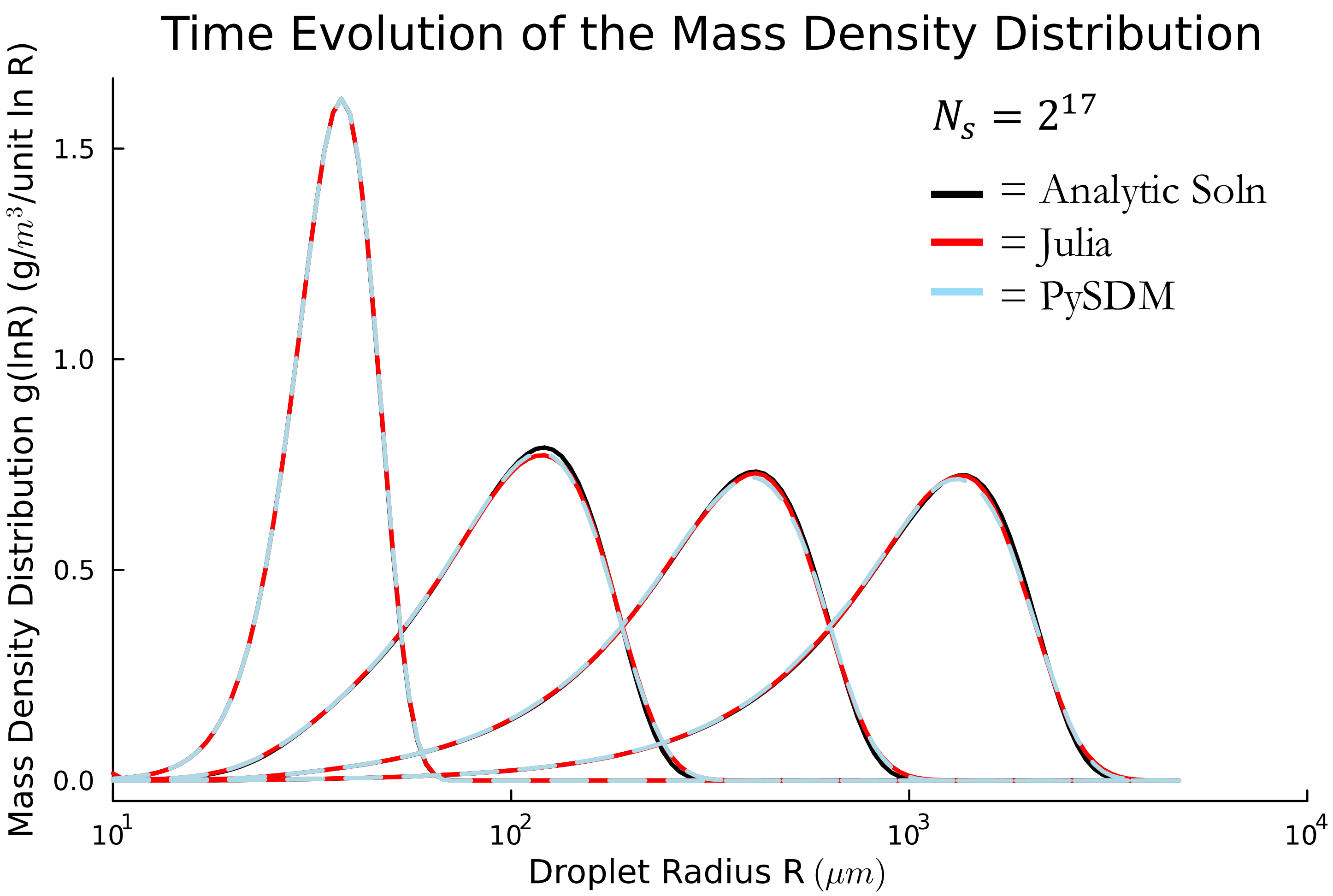
Initialization method parameterizing spread, proposed by Matsushima et al., 2023

Choose α so that distribution a that optimizes the range while keeping mean field resolution



- Defining the Deficit Problem/Adaptive Timestepping
- **Adaptivity in a Box model**
- Adaptivity in 2d prescribed flow

Box Model Test Setup



Time Evolution of the Mass Density Distribution

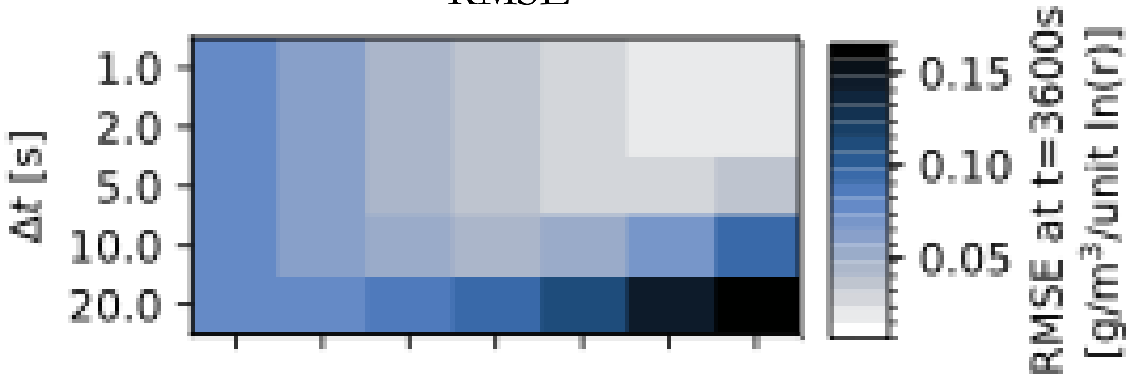
PySDM

Droplets.jl

Shima et al., (2009), Safranov (1962), Golovin (1963)

SDM Adaptivity in a Box Model

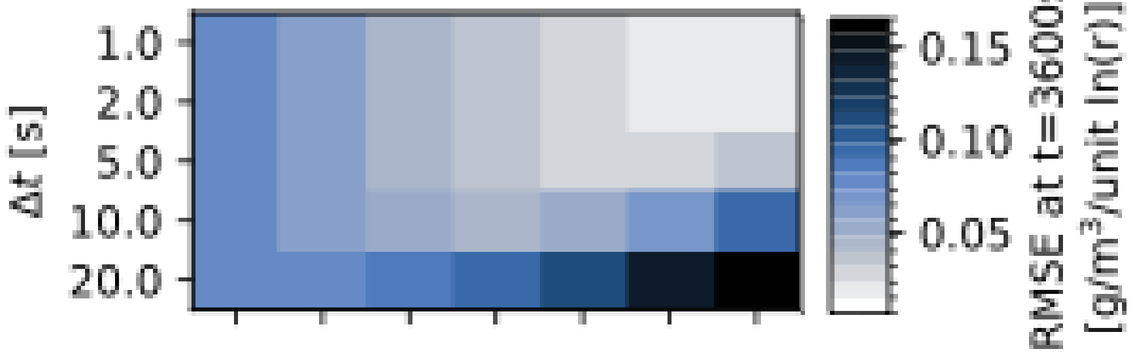
RMSE



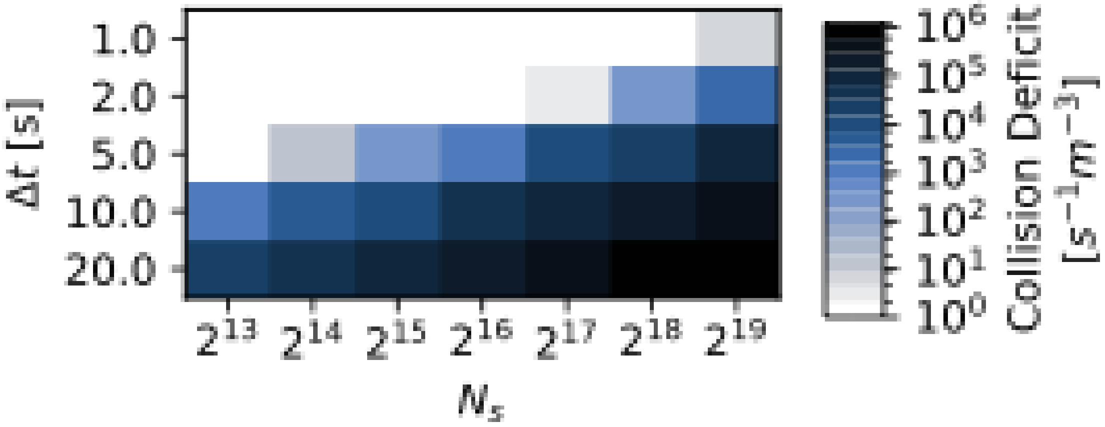
- 15 runs
- Increasing error with more superdroplets??

SDM Adaptivity in a Box Model

RMSE

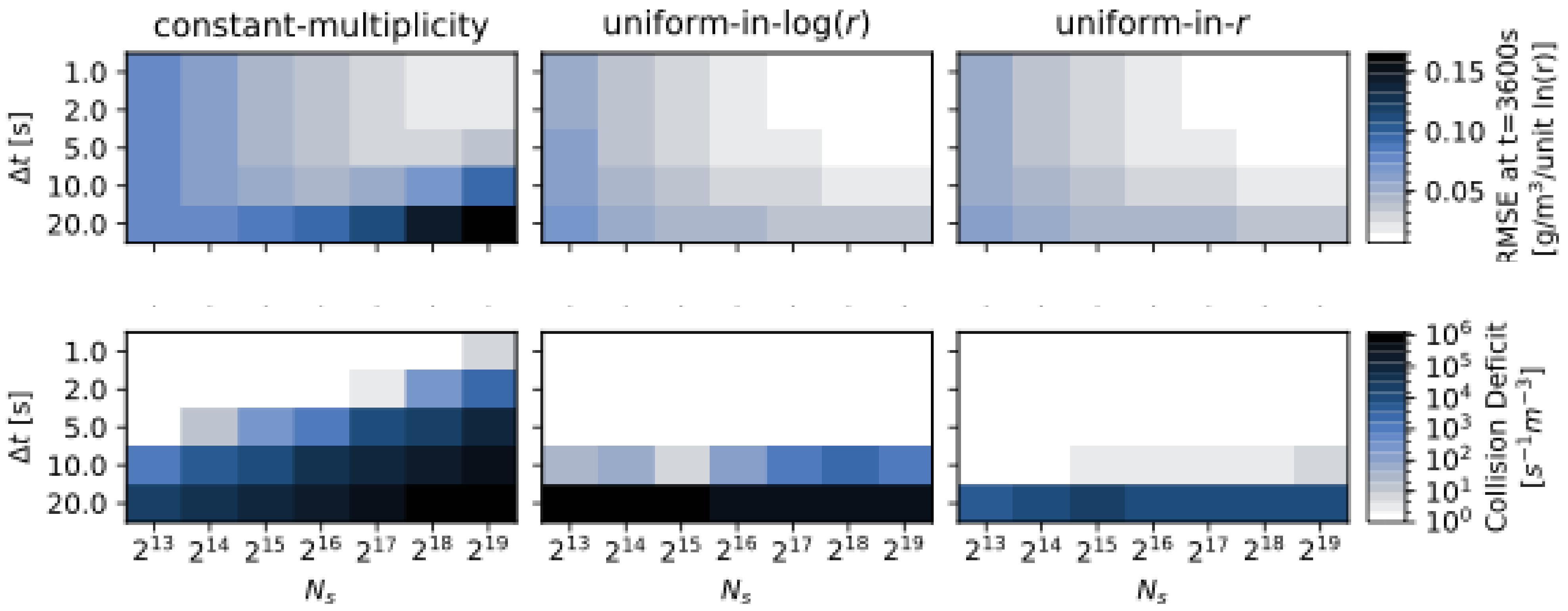


Deficit



(RMSE t =3600s)

Deficit (#/s)



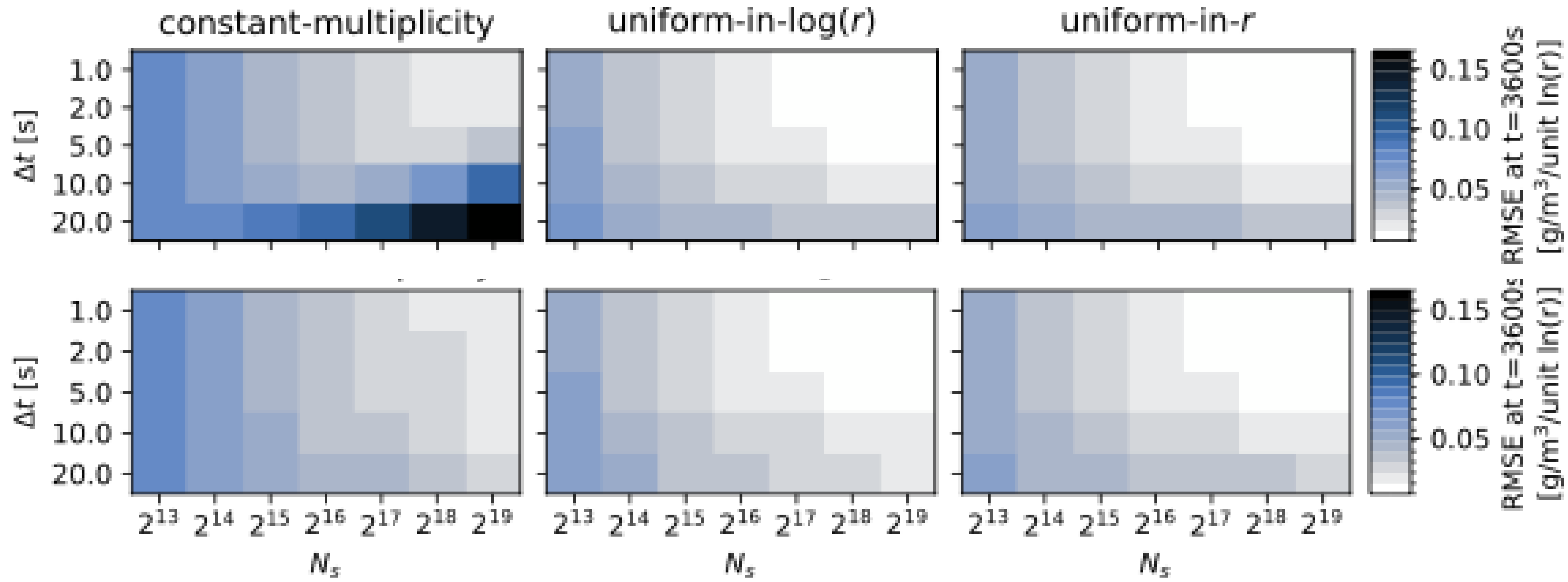
constant-multiplicity

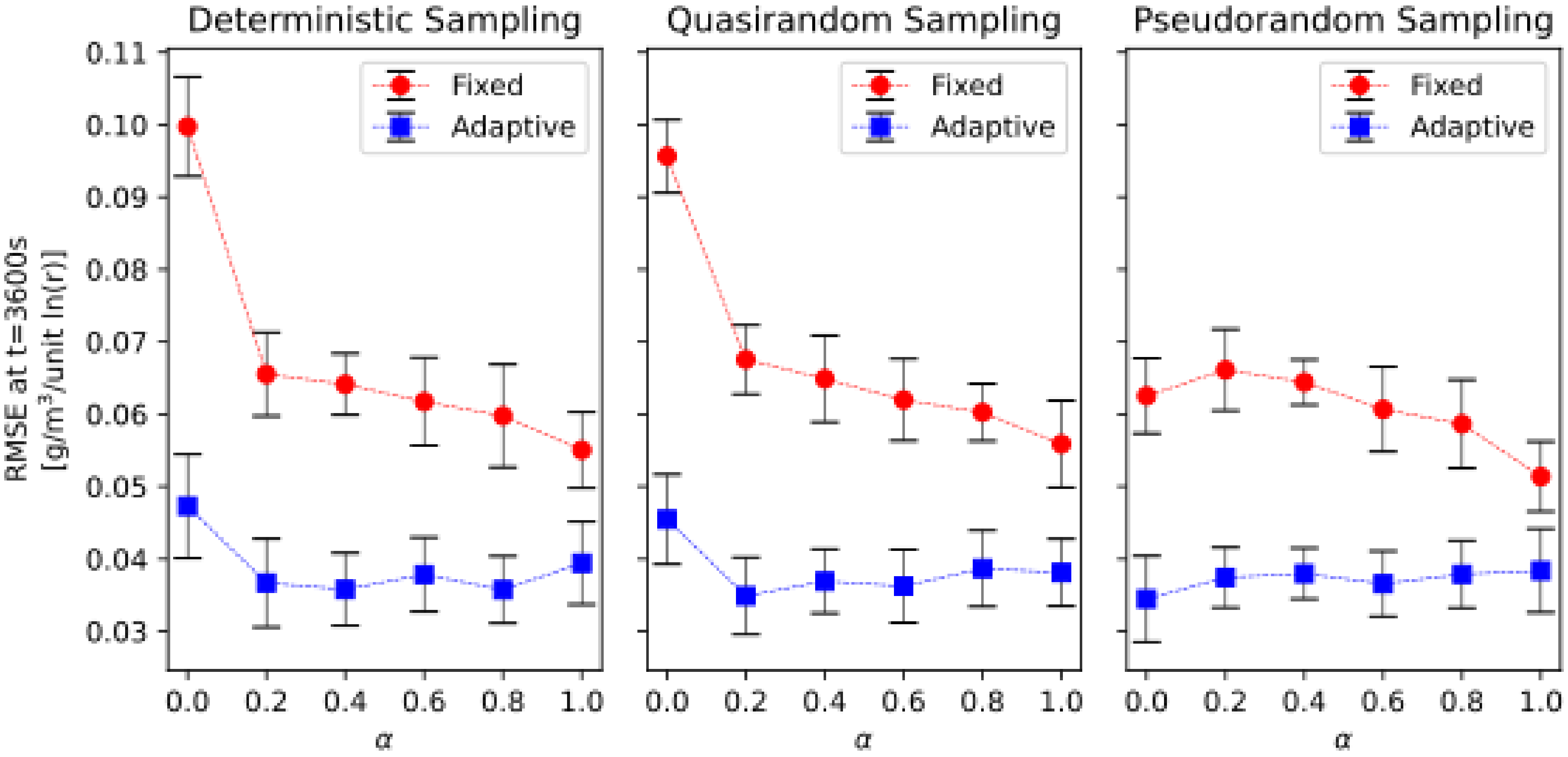
uniform-in-log(r)

uniform-in-r

No Adaptivity

Adaptivity





Deterministic Sampling
Quasirandom Sampling
Pseudorandom Sampling
Fixed
Adaptive
RMSE at t=3600s
[g/m³/unit ln(r)]
0.11
0.10
0.09
0.08
0.07
0.06
0.05
0.04
0.03
0.0
0.2
0.4
0.6
0.8
1.0
α

- Defining the Deficit Problem/Adaptive Timestepping
- Adaptivity in a Box model
- **Adaptivity in 2d prescribed flow**

2D prescribed velocity field

- Kessler 1969 Kinematic framework
- How does it behave with flow, grid exchange
- Unterstrasser et al. (2019) show less SD needed for 1d convergence
- 35 runs
- The results here only show varying time step
- 1.5x1.5 km domain, 50x50 grid, 40SD/cell
- Initialized by Uniform Log(dry radius)
- Geometric Kernel

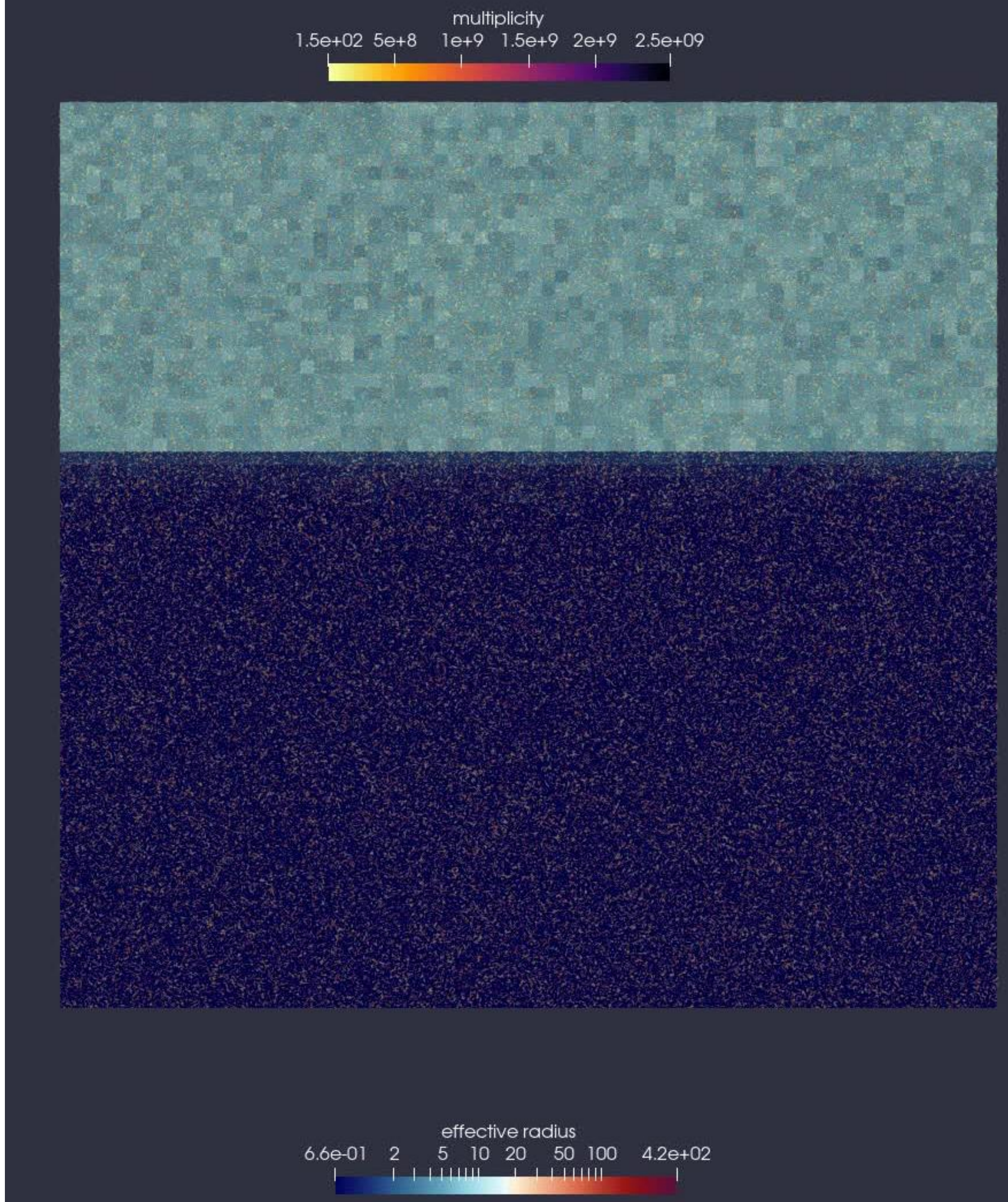
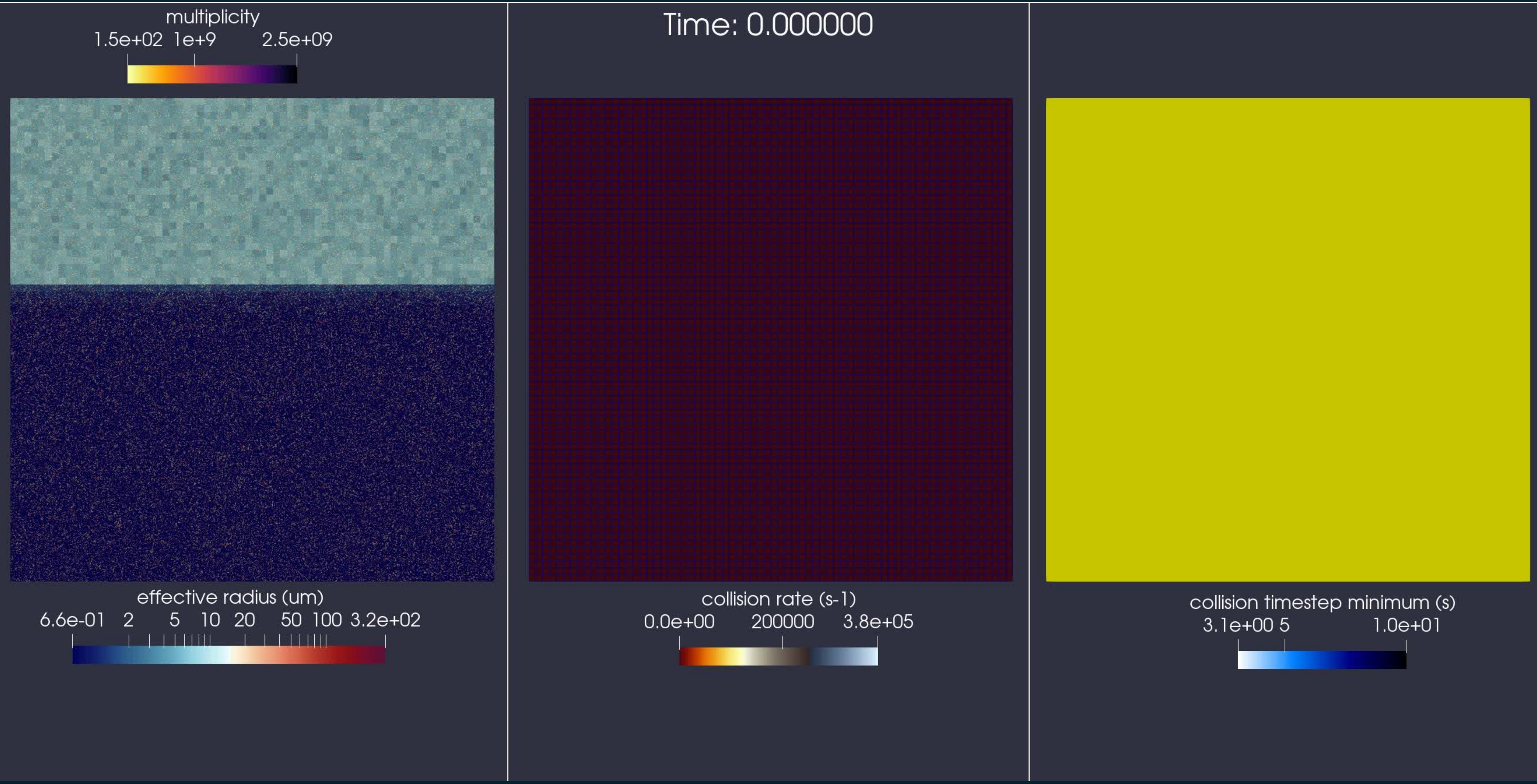
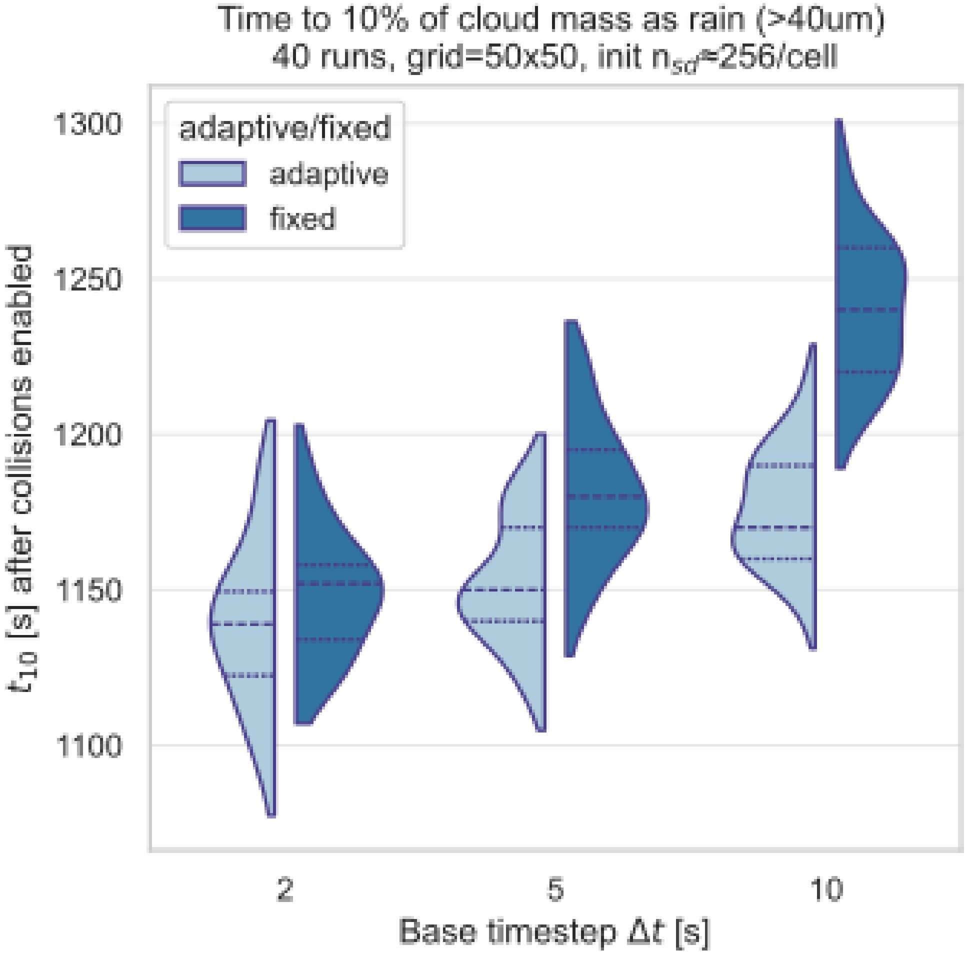
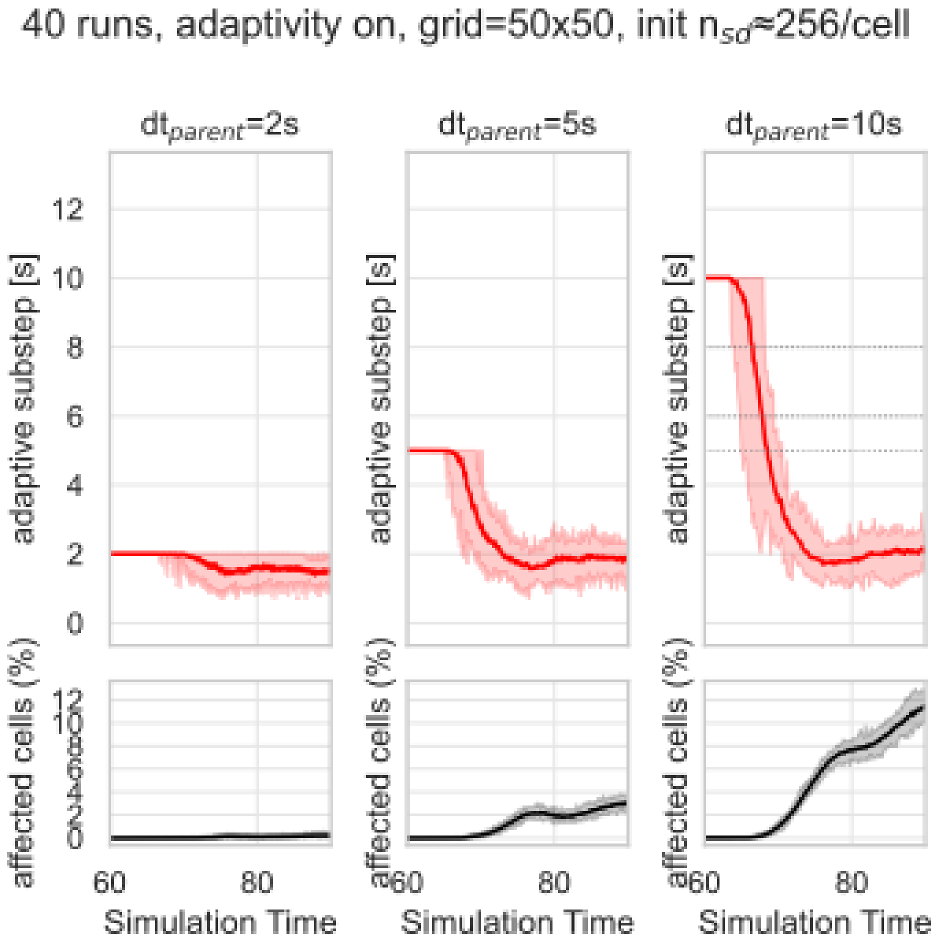


Fig. help thanks to Ola Strząbała


multiplicity
1.5e+02
1e+9
2.5e+09
Time: 0.000000
effective radius (um)
6.6e-01
2
5
10
20
50
100
3.2e+02
collision rate (s-1)
0.0e+00
200000
3.8e+05
collision timestep minimum (s)
3.1e+00
5
1.0e+01



Time to 10% of cloud mass as rain (>40um)
40 runs, grid=50x50, init n_{sd}≈256/cell



40 runs, adaptivity on, grid=50x50, init n_{sd}≈256/cell

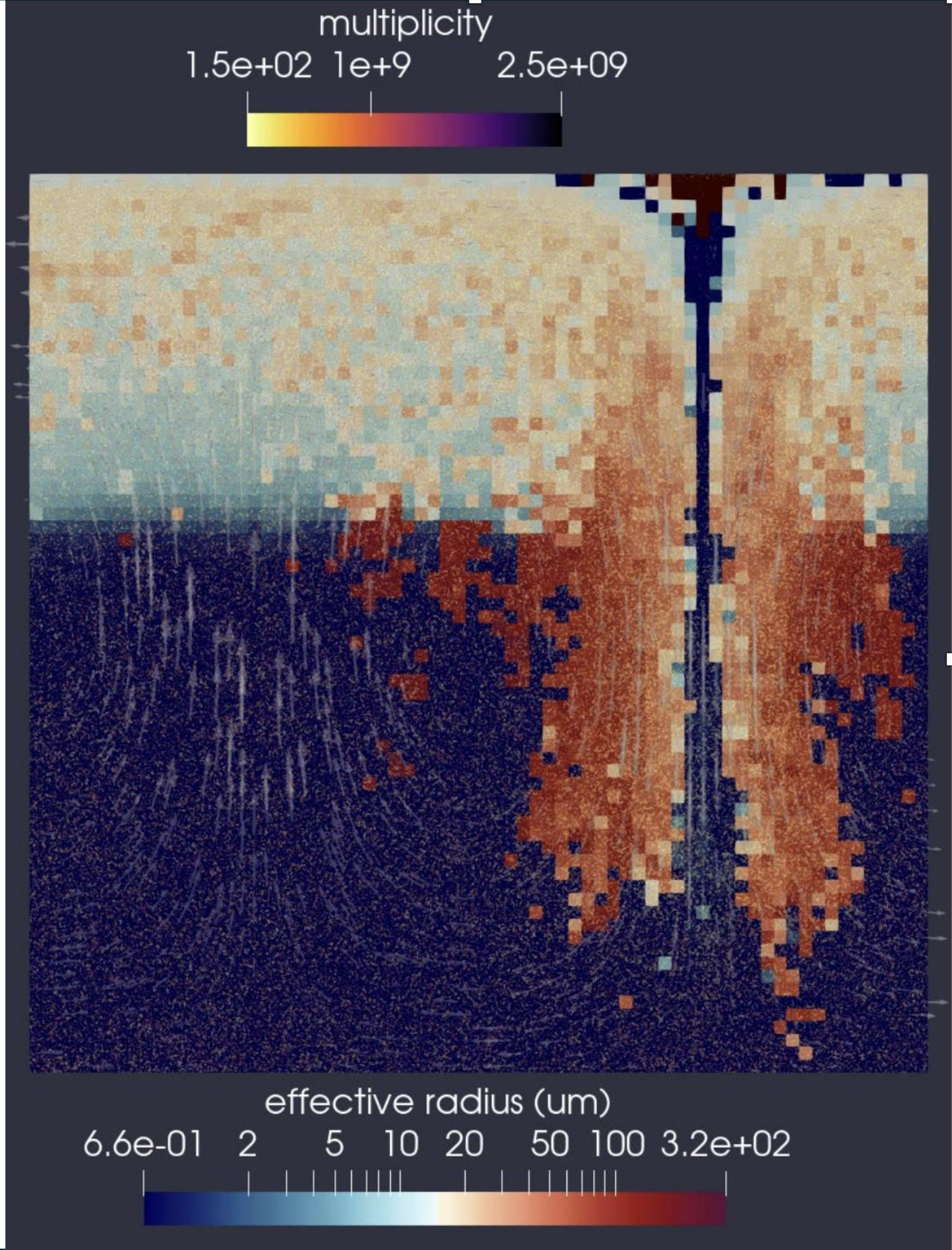
We *recommend implementation*

- The setups used might not indicate any urgent need to reduce the deficit, but embracing the adaptivity makes the scheme more robust and less sensitive to arbitrary user settings
- Easily quantified nonphysical error that only lowers the growth
- Only affecting a small fraction of the grid cells (<1%) – not a reason to lower timestep
- This is not a domain wide setting: extra computational effort is focused where needed
- Increased superdroplet resolution does not reduce deficit

arXiv:2509.05536

In summary…

- Dynamic multiplicity range helps convergence *and* deficit while increasing Nsd does not
- Adaptive Timestepping reduces sensitivity to initialization methods
- In some conditions, the deficit can affect bulk cloud properties.
- Adaptive Timestepping is the only way to eliminate deficit, and focuses computational effor where needed.
- open source science!
- Ware, Bartman et al. 2026 (submitted)


multiplicity
1.5e+02
1e+9
2.5e+09
effective radius (um)
6.6e-01
3.2e+02