University of Warsaw Lagrangian Cloud Model (UWLCM)

model formulation and stratocumulus simulations

Piotr Dziekan, Maciej Waruszewski, Hanna Pawłowska
Institute of Geophysics, Faculty of Physics, University of Warsaw

Sylwester Arabas, Anna Jaruga

Kobe, 11.2018

Agenda

- Model formulation
  - model equations
  - numerical implementation
- Stratocumulus simulations
  - 2D results
  - 3D results
- Cumulus simulations
- Chemistry ad
- Summary

LES model with SDM microphysics

EULERIAN

- Eulerian variables: θ, q_v, u, v, w
- anelastic approximation:

$$D_t \boldsymbol{u} = -\nabla \pi + \boldsymbol{k} B + \boldsymbol{F}_u,$$

$$D_t \theta = \frac{\theta^e}{T^e}\left(\frac{l_v}{c_{pd}} C\right) + F_\theta,$$

$$D_t q_v = -C + F_{q_v},$$

$$\nabla \cdot (\rho_d^r \boldsymbol{u}) = 0,$$

$$B = g\left[\frac{\theta - \theta^e}{\theta^r} + \epsilon\left(q_v - q_v^e\right) - \left(q_l - q_l^e\right)\right]$$

LAGRANGIAN

- Lagrangian representation of humidified aerosols, cloud droplets and rain drops
- κ-Köhler parametrisation of water activity

$$r\frac{dr}{dt} = \frac{D'_{\text{eff}}}{\rho_w}\left(1 - \frac{a_w\left(r, r_d, \kappa\right)\exp(A/r)}{\phi}\right),$$

$$a_w\left(r, r_d, \kappa\right) = \frac{r^3 - r_d^3}{r^3 - r_d^3\left(1 - \kappa\right)}$$

- “all-or-nothing” coalescence algorithm (Shima et al. 2009)

Numerics

EULERIAN

- advection with the MPDATA algorithm (libmpdata++)
- generalized conjugate residual pressure solver (libmpdata++)
- trapezoidal integration of buoyancy and pressure gradient
- Euler forward integration of other forcings
- calculated on CPUs

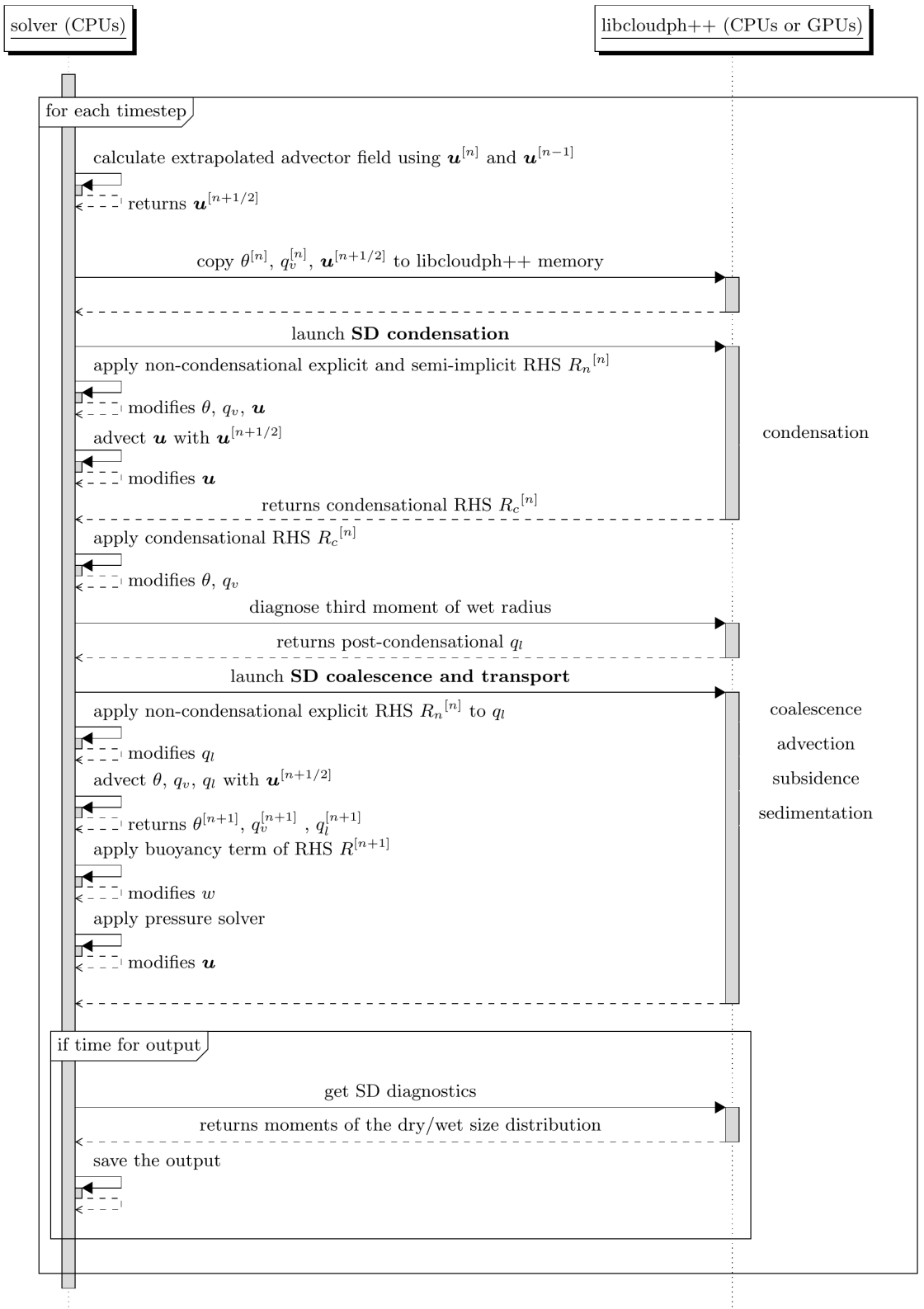
LAGRANGIAN

- Super-droplet method (libcloudph++)
- SD attributes: multiplicity, dry and wet radii, hygroscopicity
- integration of growth equation

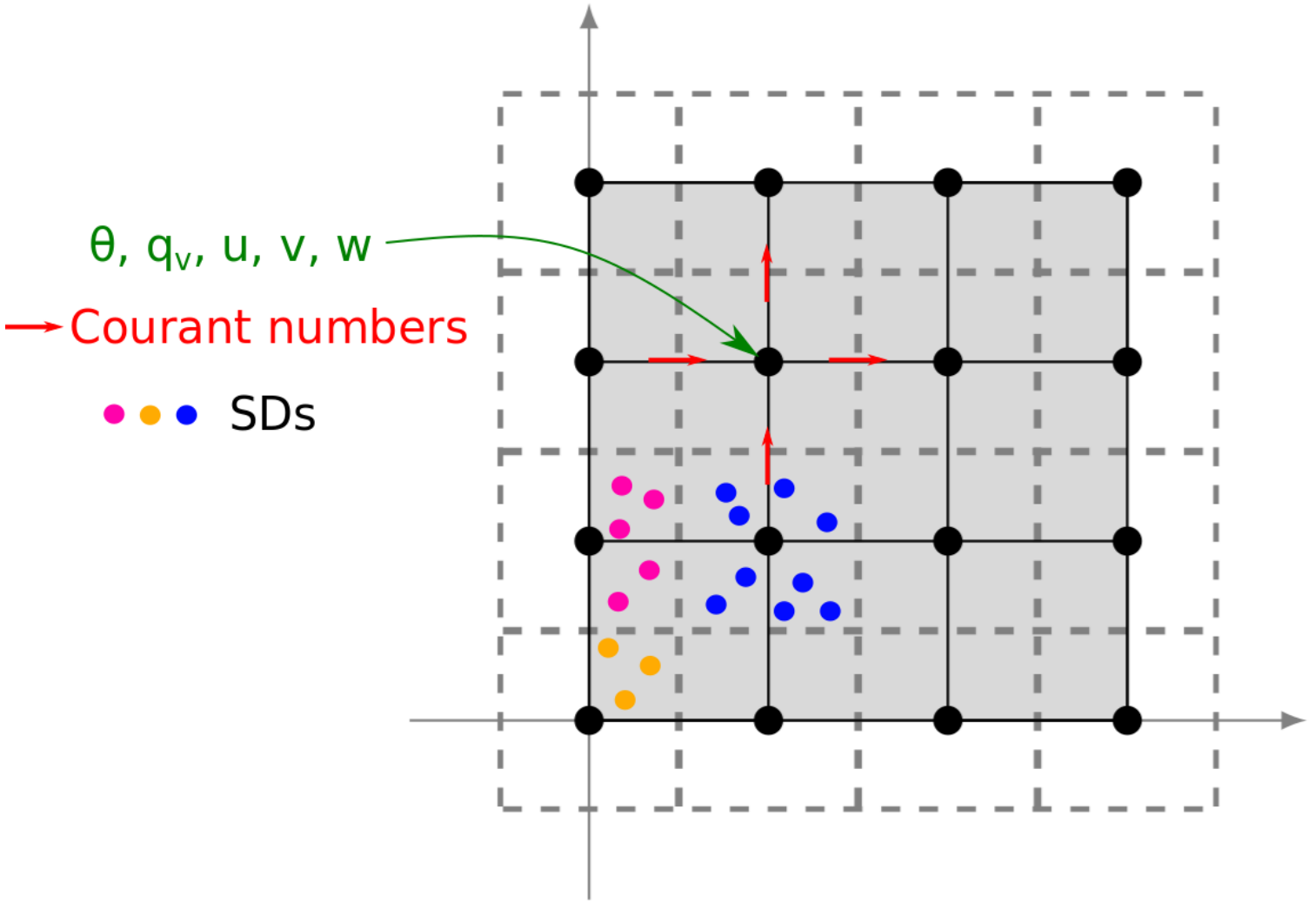
$$r^{2[n+1]} = r^{2[n]} + \Delta t \left. \frac{dr^2}{dt} \right|_{r^{2[n+1]}, q_v^{[n]}, \theta^{[n]}}$$

- predictor-corrector advection of SDs
- calculated on GPUs

Time step sequence



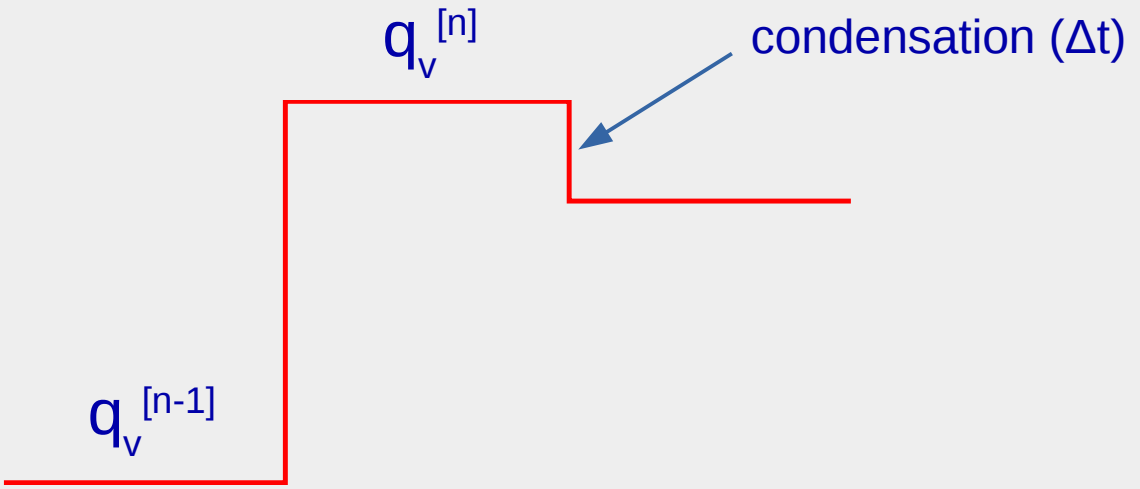
Spatial discretization



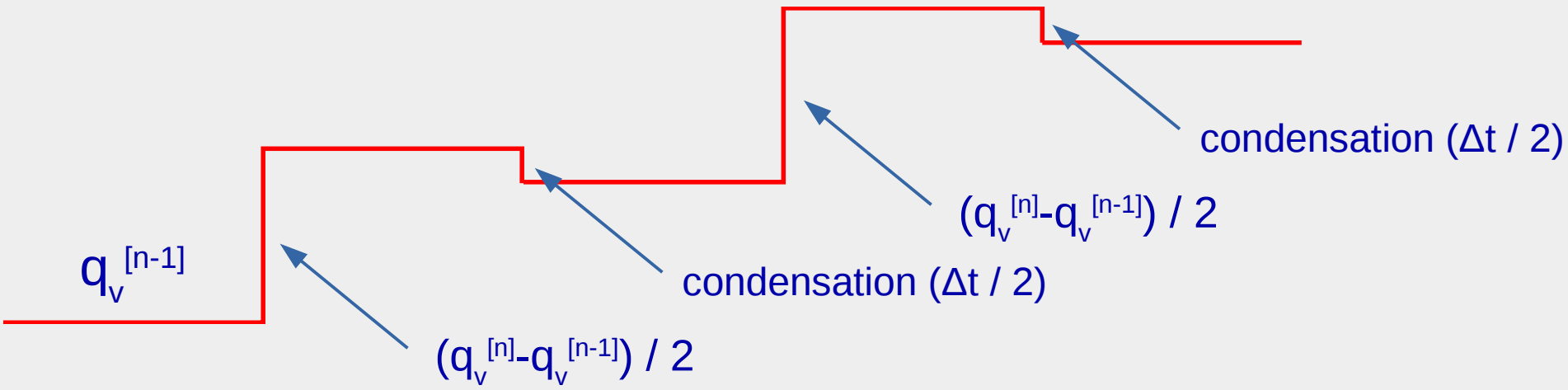
Condensation sub-stepping

- Scheme for integration of the condensation equation converges for $\Delta t \approx 0.1$ s
- LES timestep $\Delta t \approx 1$ s
- two algorithms tested:
  - *per-cell*
  - *per-particle*

no sub-stepping:



2 sub-steps:



- *per-cell*: $q_v^{[n-1]}$ comes from the cell in which the SD is at step n
- *per-particle*: $q_v^{[n-1]}$ comes from the cell in which the SD was at step n-1

Comparison with other models - DYCOMS RF02 stratocumulus simulations

- A drizzling marine stratocumulus
- Test UWLCM against 11 LES models from the Ackerman et al. 2009 intercomparison
- Microphysical schemes in other models:
  bin, single-moment bulk and double-moment bulk
- implicit LES in UWLCM, other models with explicit subgrid-scale schemes
- 2D and 3D simulations

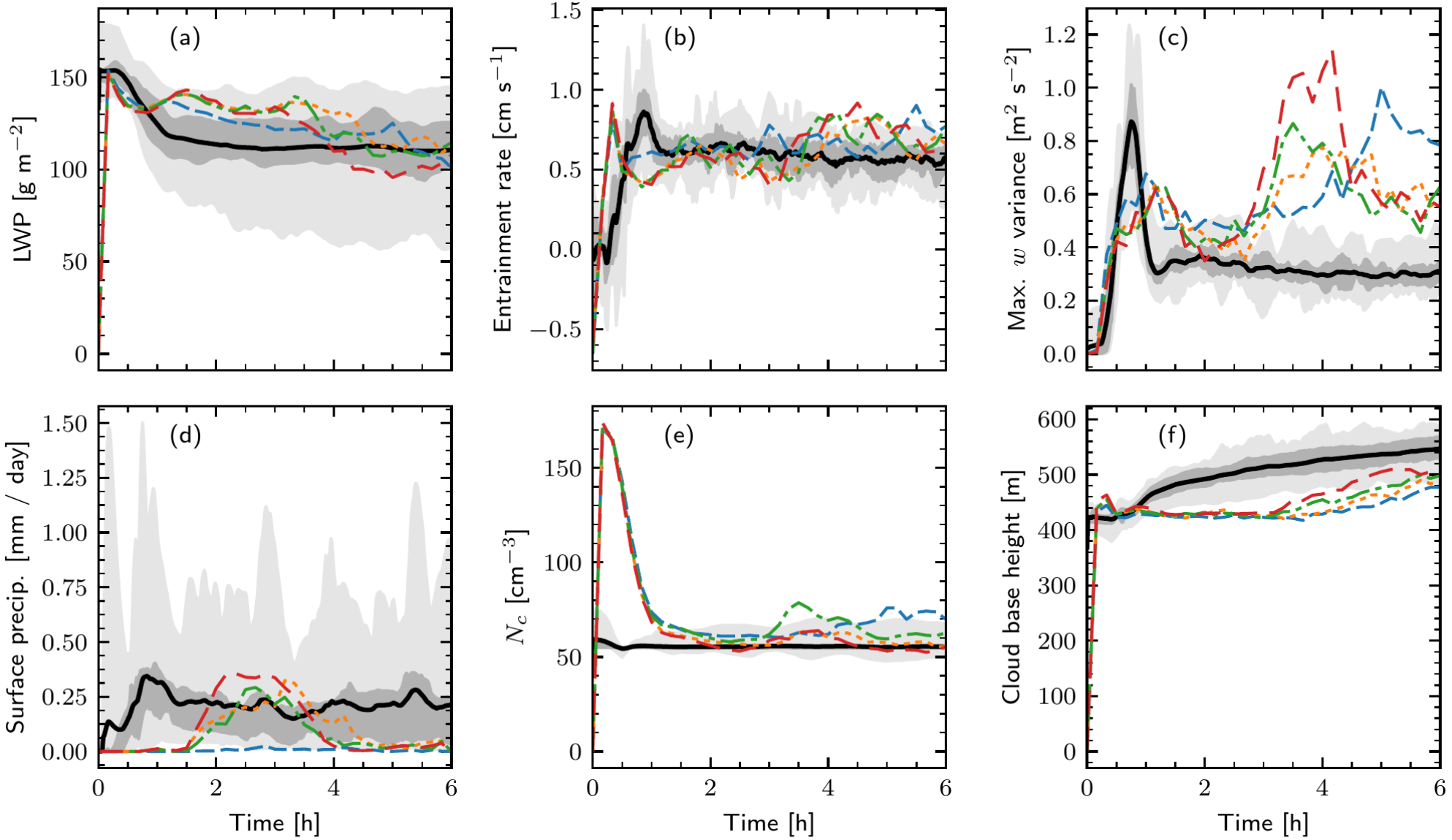
source: Angela Rowe
communitycloudatlas.wordpress.com

ANIMATION

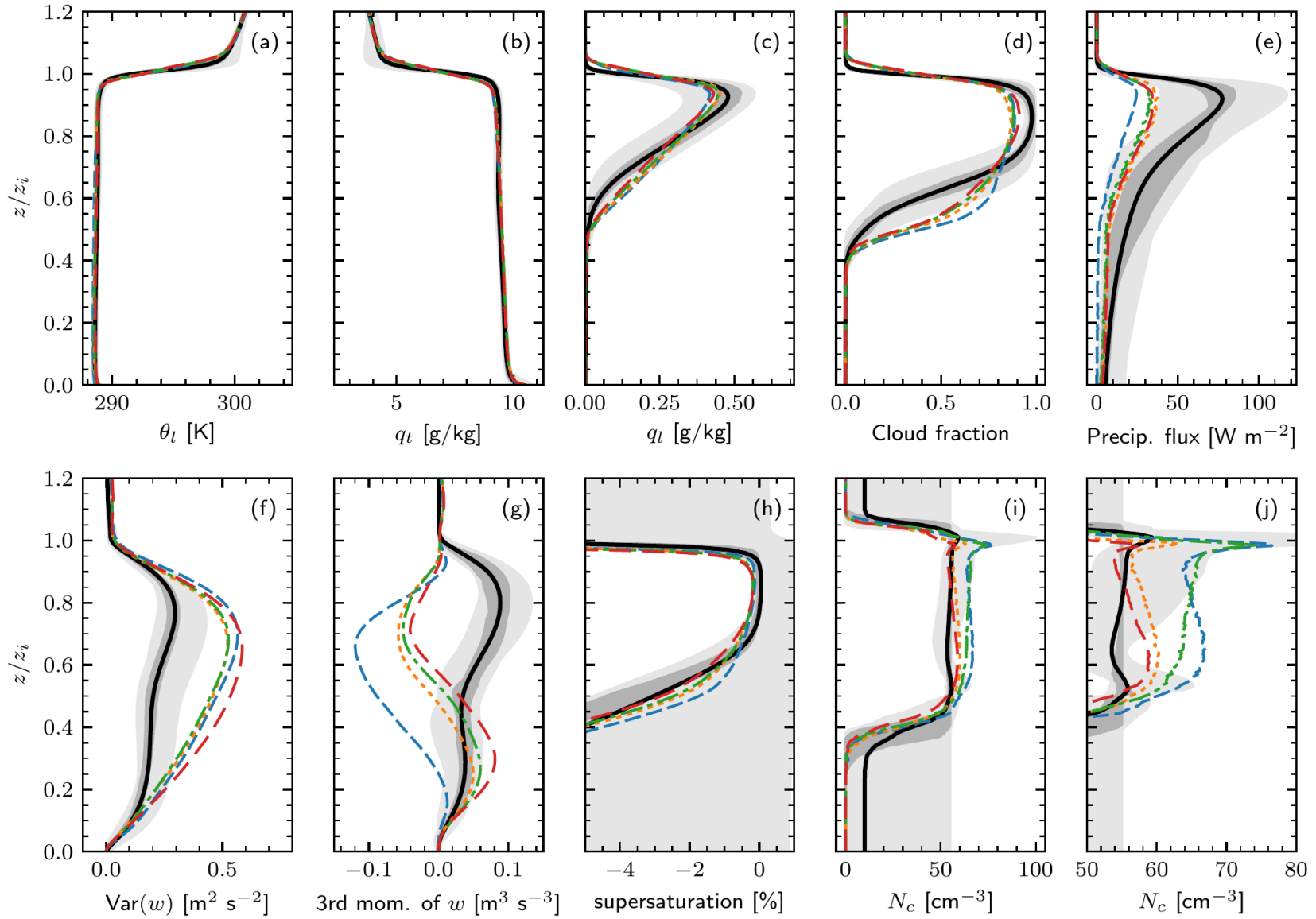
2D simulations

- Test the *per-cell* and *per-particle* sub-stepping algorithms
- Test different time step lengths
- Test different numbers of SDs,
  N_{SD} - initial number of SDs per cell
- Comparison with 3D results from Ackerman et al. 2008

2D time series



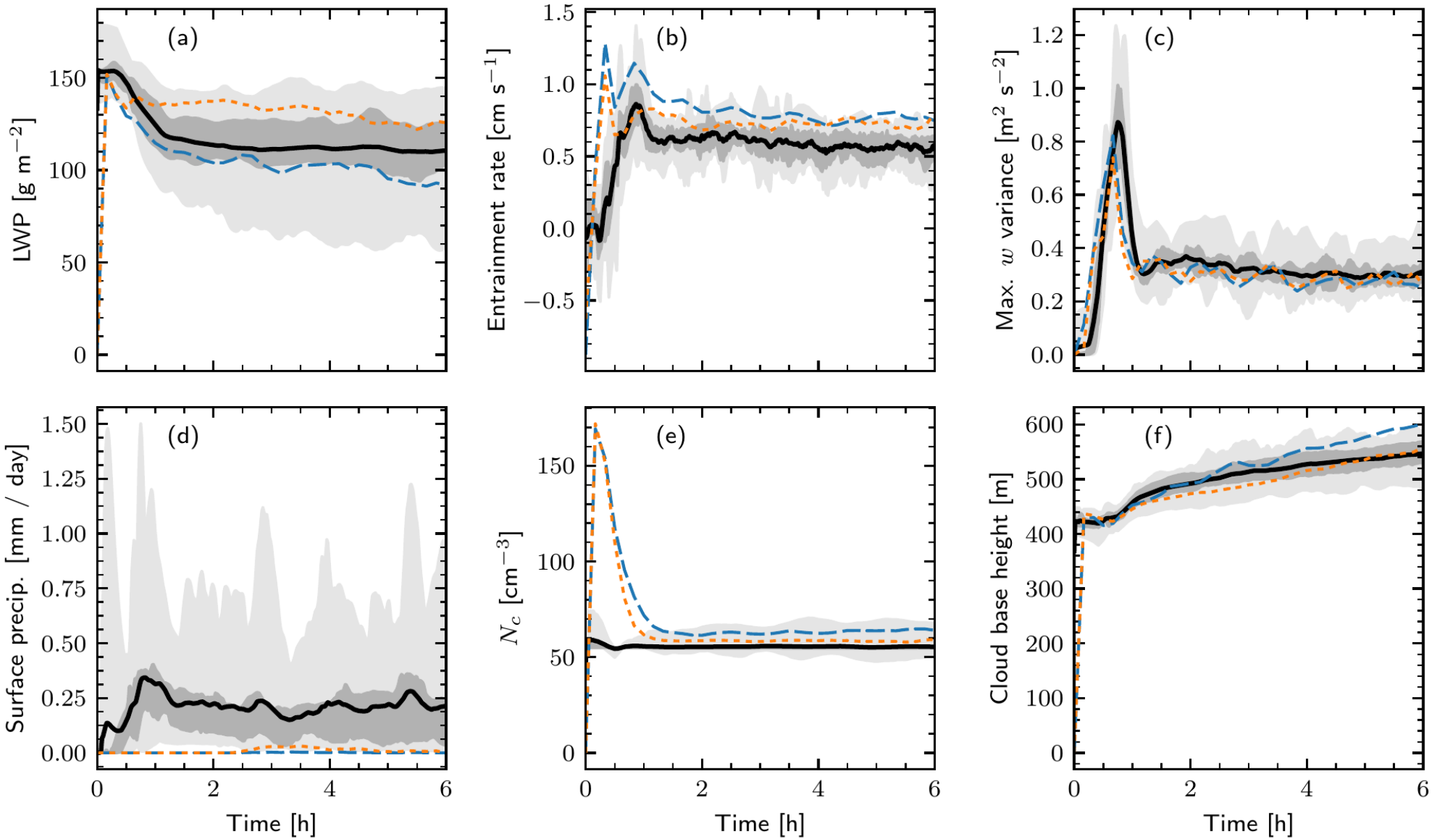
2D vertical profiles



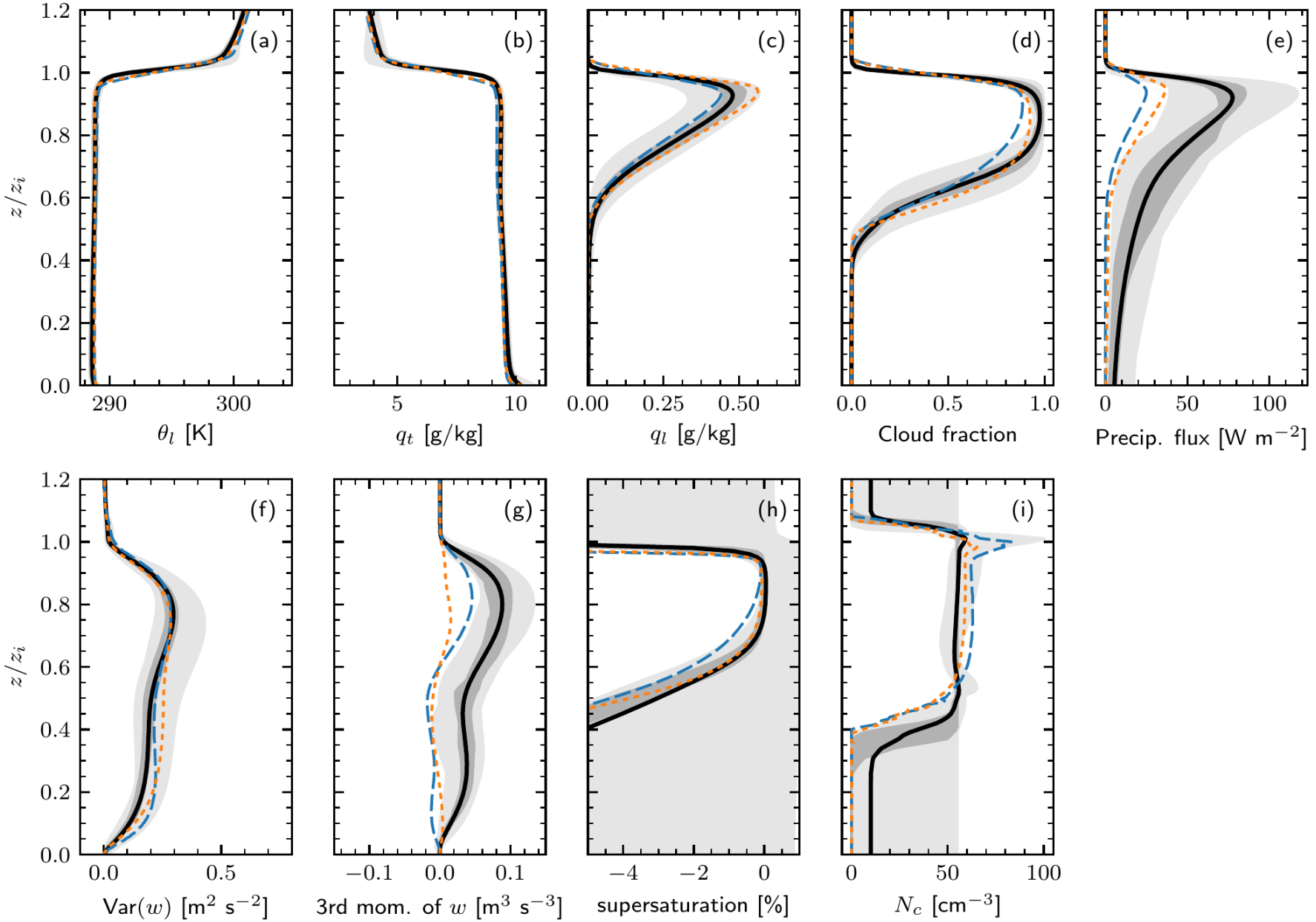
3D simulations

- Use the *per-particle* sub-stepping algorithm
- $N_{SD} = 40$
- Test for different time step lengths
- Comparison with 3D results from Ackerman et al. 2008

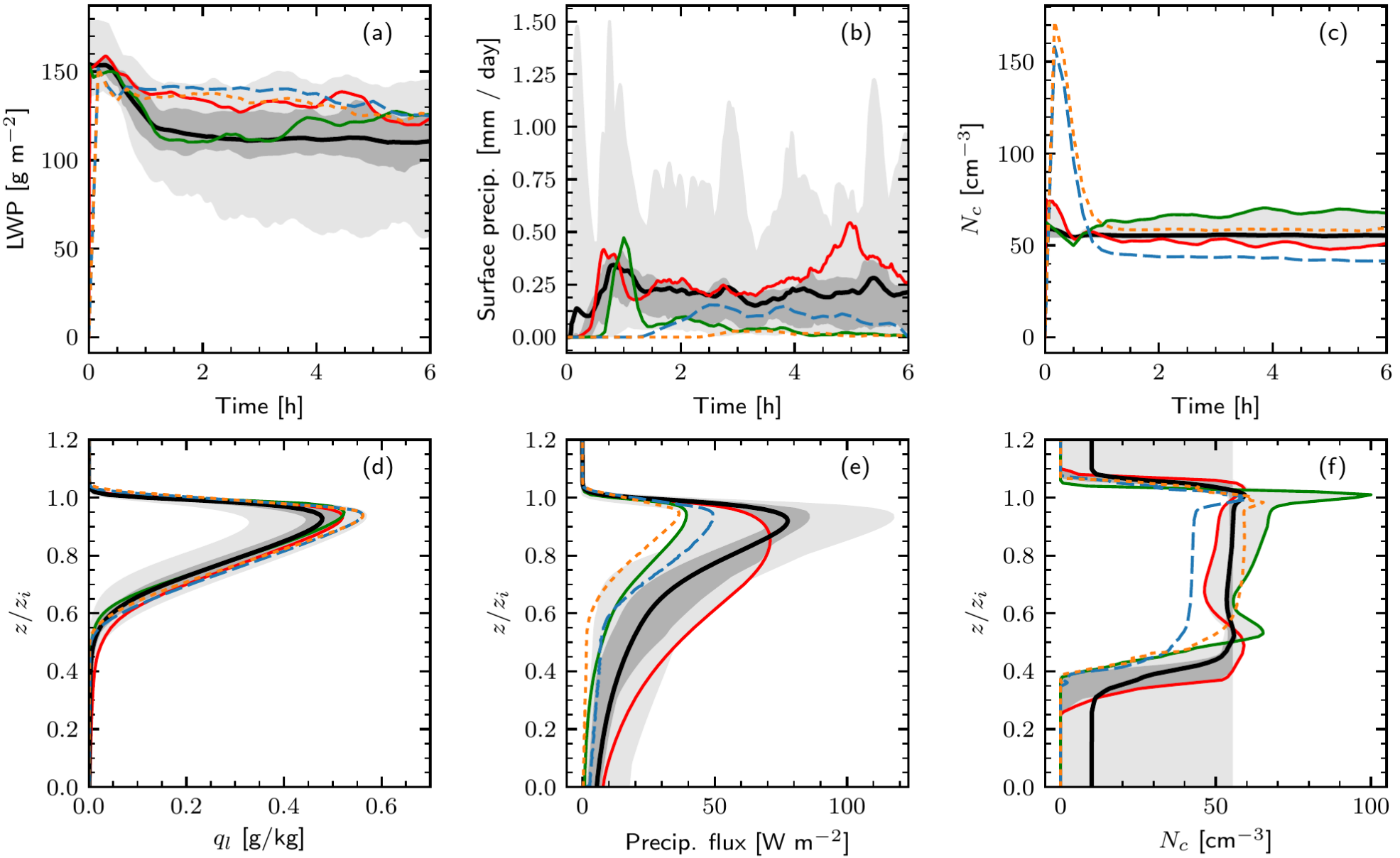
3D time series



3D vertical profiles



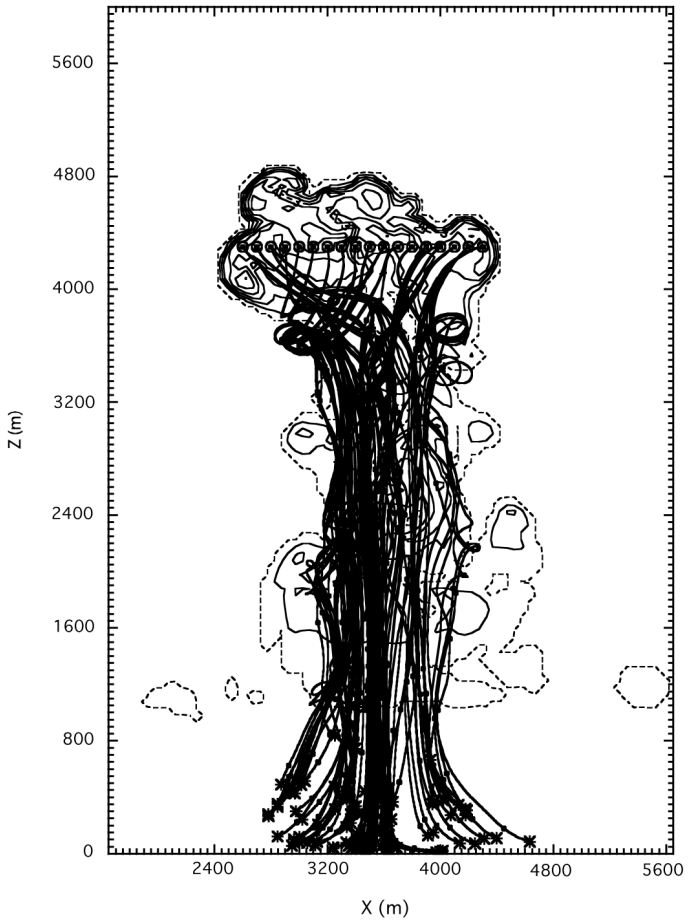
3D: comparison with bin microphysics

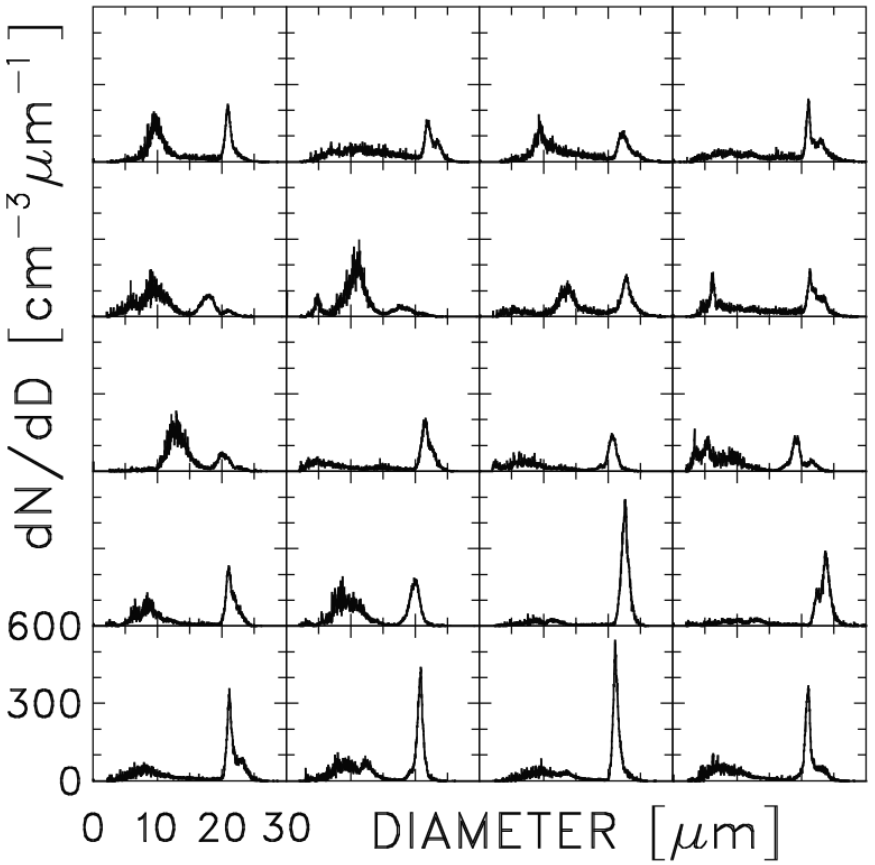


Stratocumulus modeling - conclusions

- General agreement between UWLCM and reference models
- Discrepancies in the third moment of vertical velocity (ILES vs SGS ?)
- Little rain in UWLCM
- N_{SD} = 40 same as N_{SD} = 1000
- *per-particle* sub-stepping necessary for correct activation
- 2D simulations convenient for testing and microphysics studies

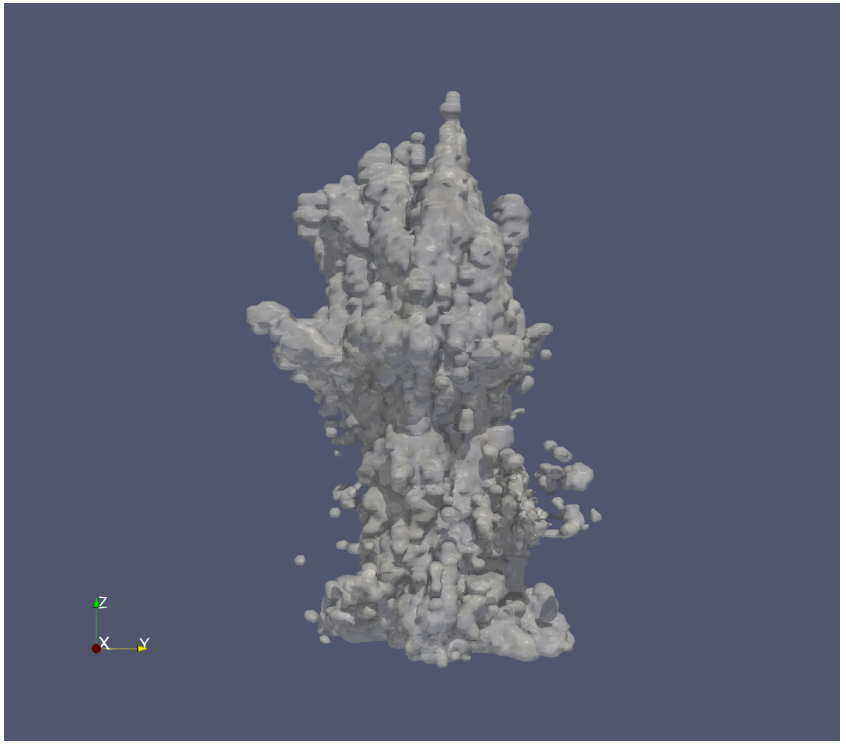
Advantages of particle-based microphysics: Cumulus simulations

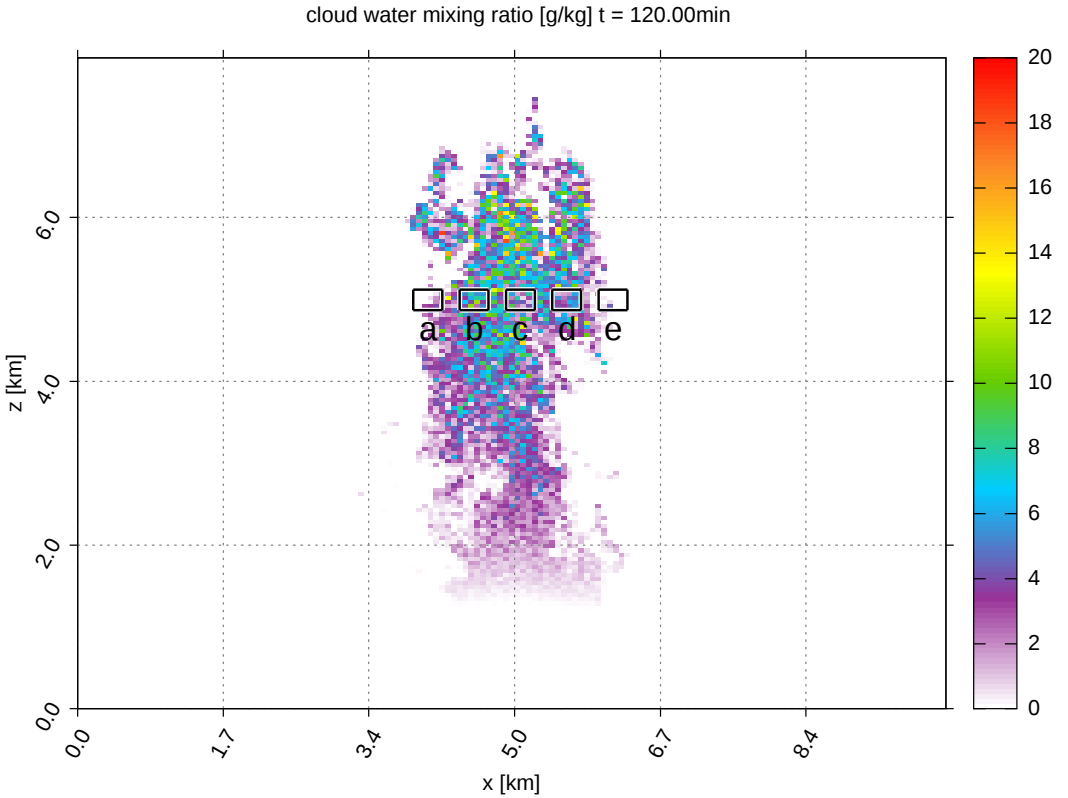


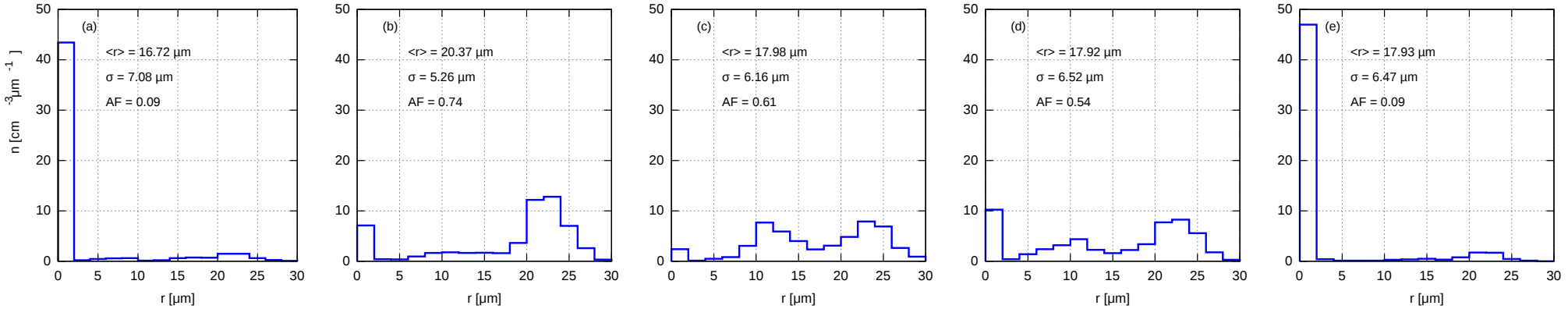


reproduced from Lasher-Trapp et al. *Q. J. R. Met. Soc.* 2005

UWLCM cumulus results







libcloudph++ 2.0: aqueous-phase chemistry extension of the particle-based cloud microphysics scheme

Anna Jaruga and Hanna Pawlowska

Institute of Geophysics, Faculty of Physics, University of Warsaw, Warsaw, Poland

Geoscientific Model Development

Geosci. Model Dev., 11, 3623–3645, 2018
https://doi.org/10.5194/gmd-11-3623-2018

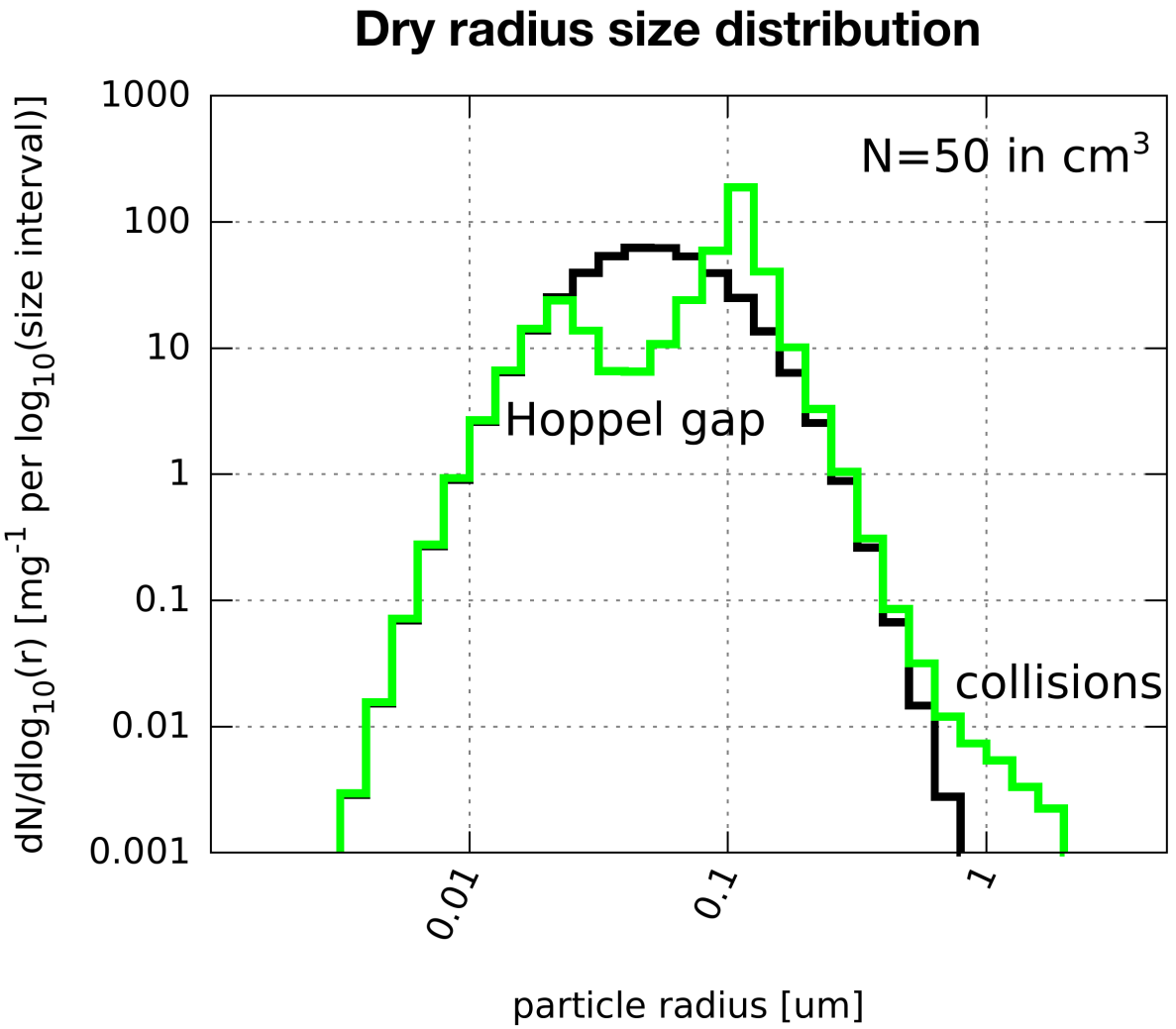
Chemical reaction: oxidation of SO_2 by O_3 and H_2O_2

Included trace gases: SO_2, CO_2, O_3, H_2O_2, NH_3, HNO_3

Processes:

- uptake of trace gases into droplets (modelled as a non-equilibrium process)
- dissociation into ions (iterative search for equilibrium pH of each super-droplet)
- oxidation of SO_2 by O_3 and H_2O_2

Example results



- Oxidation reaction is irreversible and affects the sizes of aerosol particles that served as condensation nuclei for cloud droplets
- As a result a gap in sizes between the activated and not activated aerosols is formed (Hoppel gap)
- Additionally a tail of larger aerosol particles is formed due to collisions between water drops

Summary

- Particle-based Lagrangian microphysics used in LES give stratocumulus results in agreement with bulk and bin microphysics, except for lower precipitation.
- Implicit LES is in agreement with LES with subgrid-scale models, with the exception of skewness of the vertical velocity distribution.
- Thanks to the use of GPUs, sophisticated microphysics do not slow down simulations.
- LES code available for everyone: https://github.com/igfuw/UWLCM
- manuscript submitted to GMD

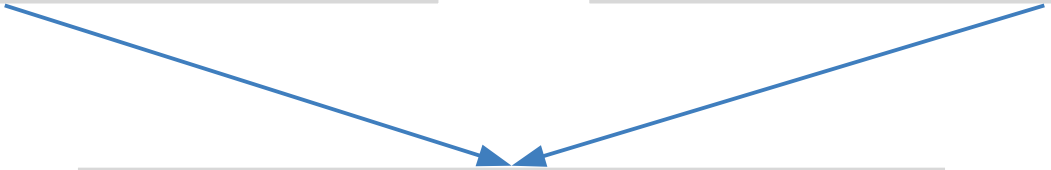
Software developed by our group

libmpdata++

- new implementation of MPDATA
- written in C++
- started ca. 5 years ago

libcloudph++

- cloud microphysics routines
- written in C++ with python bindings
- started ca. 5 years ago



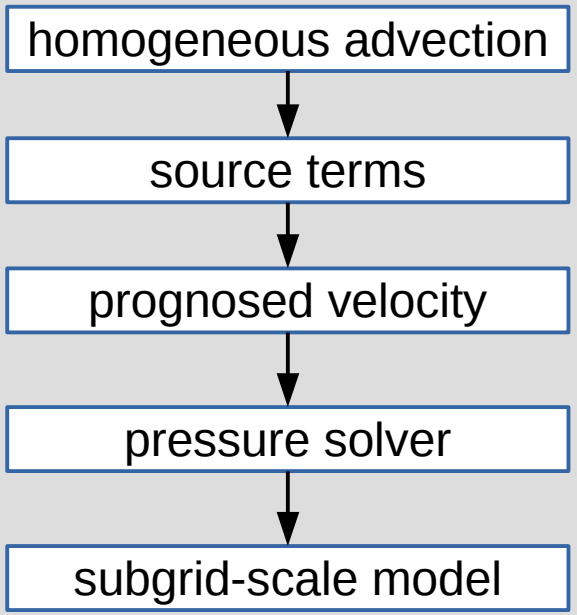
UWLCM

- Large Eddy Simulations of clouds using anelastic approximation
- written in C++
- started ca. 2 years ago

Modern code structure: separation of concerns

libmpdata++

- hierarchy of solvers:

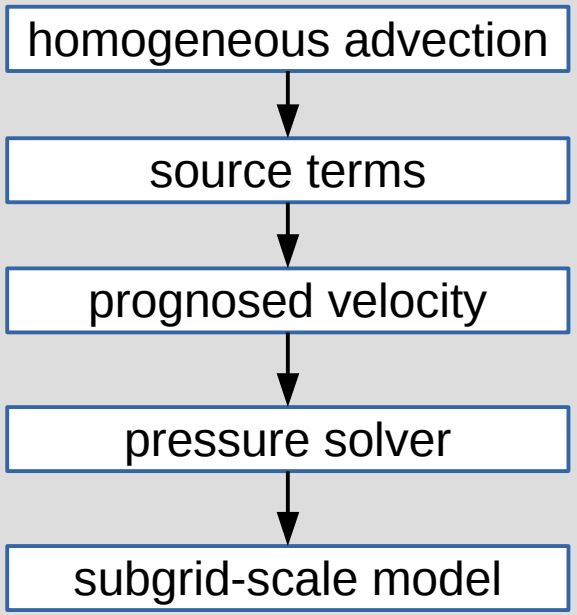


- boundary conditions
- concurrency handlers
- output handlers

Modern code structure: separation of concerns

libmpdata++

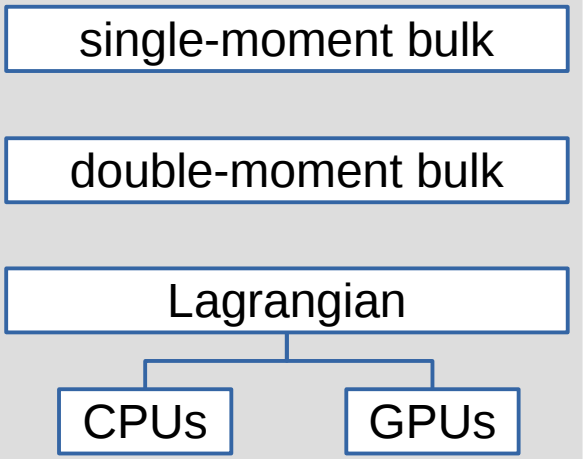
- hierarchy of solvers:



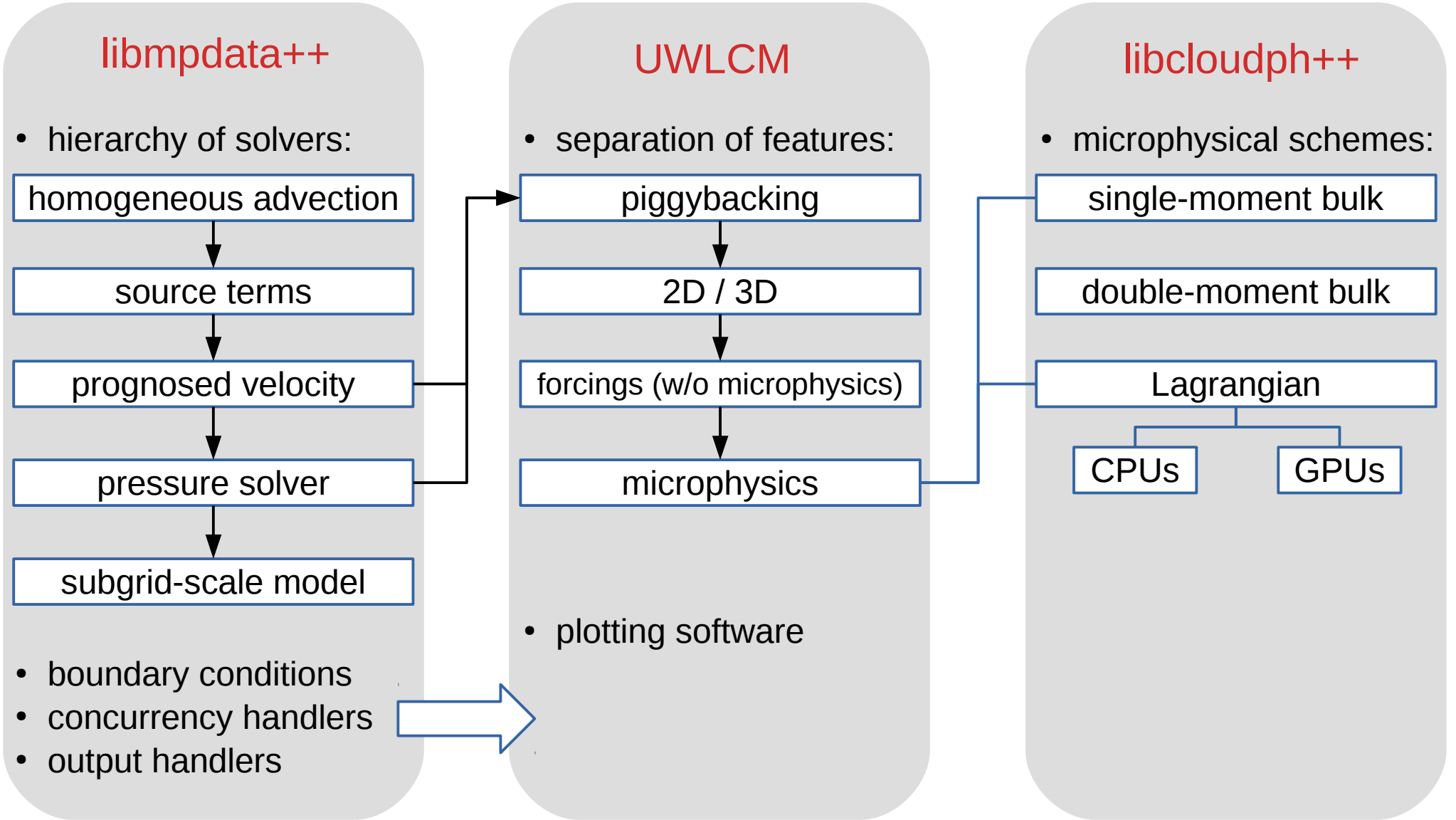
- boundary conditions
- concurrency handlers
- output handlers

libcloudph++

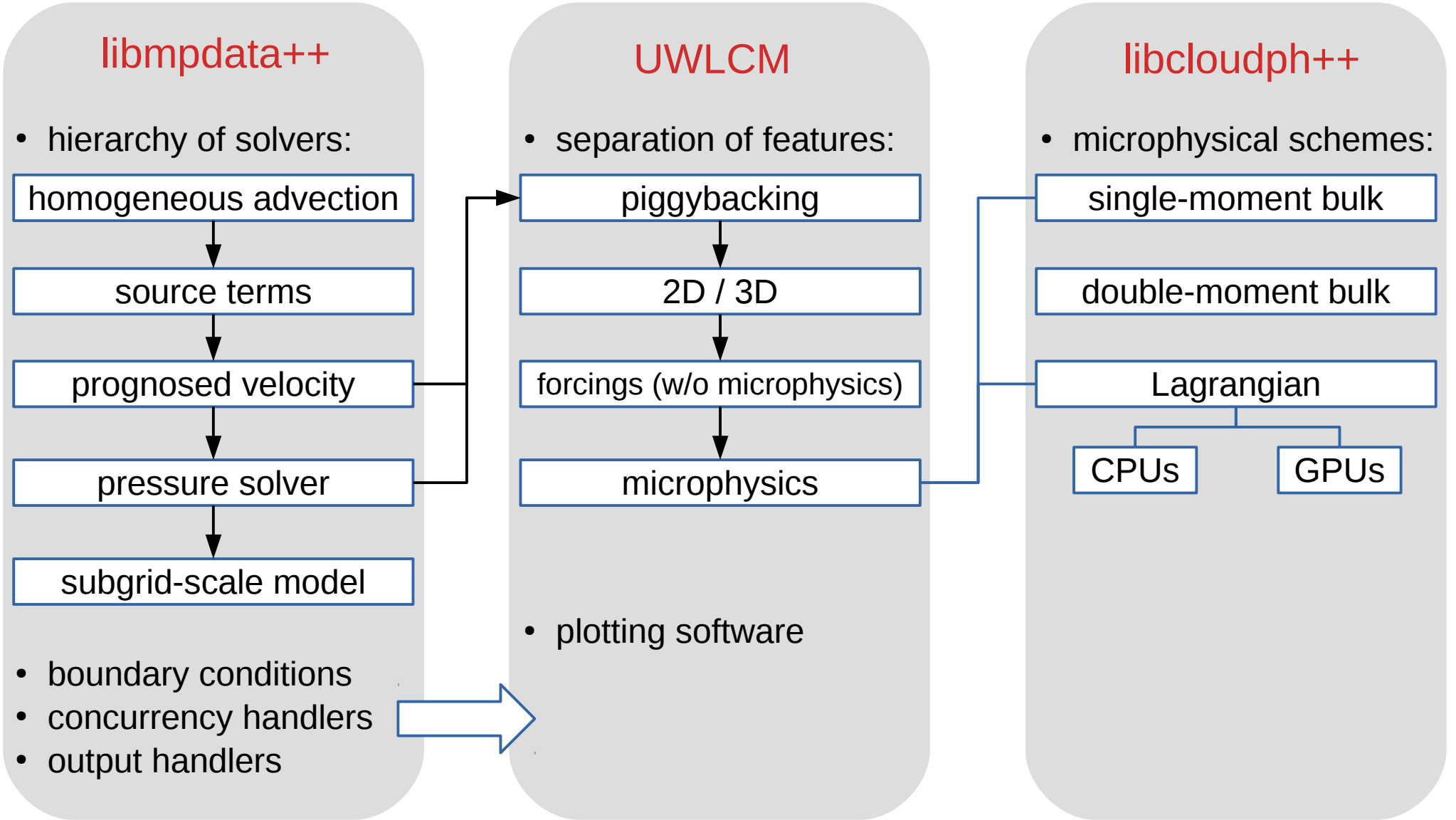
- microphysical schemes:



Modern code structure: separation of concerns



Modern code structure: separation of concerns



- Code sections can be developed independently.
- Code are sections ready to be reused.

- version control system
- automated tests
- open-source code hosted on github

Benefits of the modern code structure: one code, many models

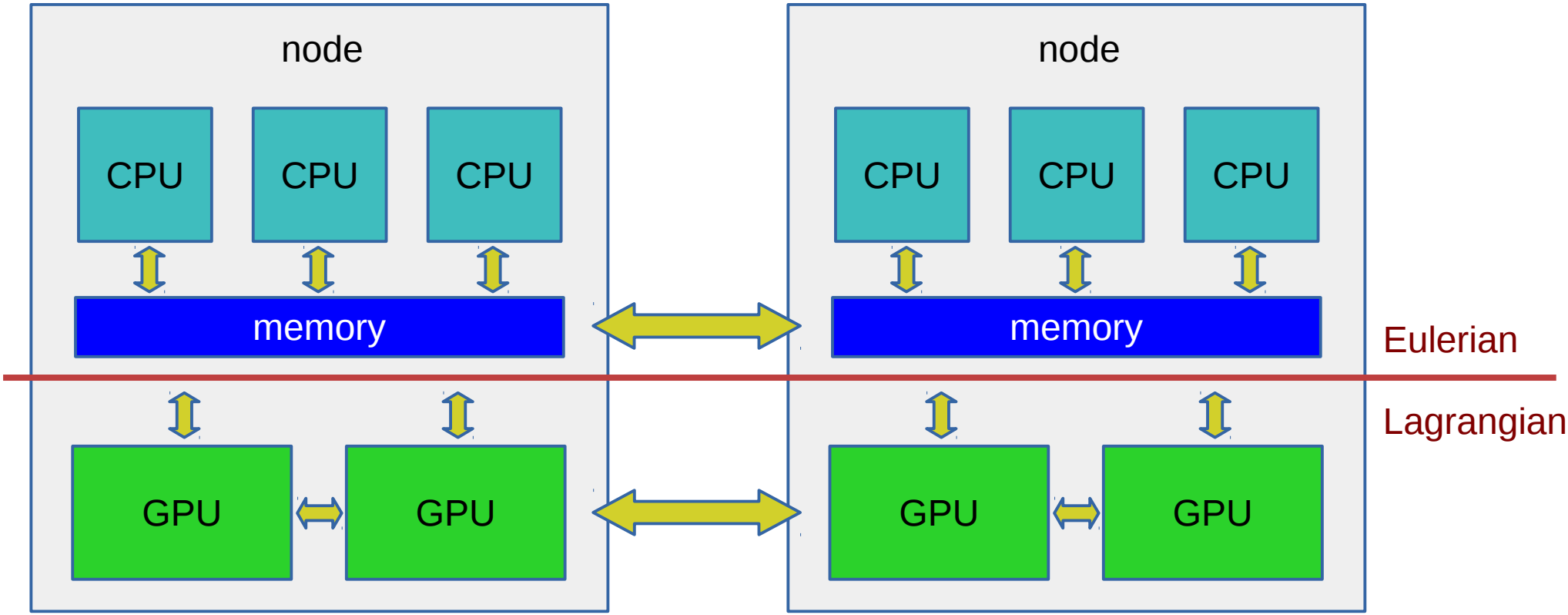
runtime options:

- number of dimensions:
  ./bicycles --ny=[0, X]
- type of microphysics:
  ./bicycles --micro=[blk_1m, lgrngn]
- where to calculate Lagrangian microphysics:
  ./bicycles --backend=[serial, OpenMP, CUDA, multi_CUDA]
- model setup:
  ./bicycles --case=[dycoms, bomex, ...]
- piggybacking:
  ./bicycles --piggy=1 --vel_in=file
- number of CPU threads for dynamics:
  OMP_NUM_THREADS=X ./bicycles
- distributed memory runs:
  mpiexec -np X ./bicycles
- advection, coalescence, condensation timesteps
- number of super-droplets
- ...

compile time options:

- MPDATA options
  - variable-sign option:
    opts = [opts::iga, opts::abs]
  - non-oscillatory option:
    opts = [opts::fct]
  - third-order terms:
    opts = [opts::tot]
  - ...
- microphysics options
  - coalescence kernel
  - terminal velocity
  - ....
- numerical precision
  real_t = [float, double]

Modern HPC architecture



- Domain decomposition between nodes
- Separate intra-node domain decomposition for CPU threads and GPUs
- Bulk microphysics computed on CPUs
- Lagrangian microphysics computed on CPUs or GPUs
- Simultaneous computations of fluid flow on CPUs and microphysics on GPUs synchronized only during condensation