

# Lagrangian warm-rain microphysics in libcloudph++

Sylwester Arabas, Anna Jaruga, Piotr Dziekan  
Hanna Pawłowska, Wojciech Grabowski

Institute of Geophysics  
Faculty of Physics, University of Warsaw

Workshop on Eulerian vs. Lagrangian methods for cloud microphysics  
Warsaw, April. 20<sup>th</sup> 2015

## project architects & goals

Hanna Pawłowska, Piotr Smolarkiewicz & Wojtek Grabowski

develop a new LES tool for aerosol-cloud interactions research

# project architects & goals

Hanna Pawłowska, Piotr Smolarkiewicz & Wojtek Grabowski

develop a new LES tool for aerosol-cloud interactions research



- ▶ CCN activation
- ▶ condensational growth

# project architects & goals

Hanna Pawłowska, Piotr Smolarkiewicz & Wojtek Grabowski

develop a new LES tool for aerosol-cloud interactions research



- ▶ CCN activation
- ▶ condensational growth



- ▶ collisional growth
- ▶ aqueous chemistry

# project architects & goals

Hanna Pawłowska, Piotr Smolarkiewicz & Wojtek Grabowski

develop a new LES tool for aerosol-cloud interactions research



- ▶ CCN activation
- ▶ condensational growth



- ▶ collisional growth
- ▶ aqueous chemistry



- ▶ precipitation
- ▶ wet deposition
- ▶ droplet deactivation

# project architects & goals

Hanna Pawłowska, Piotr Smolarkiewicz & Wojtek Grabowski

develop a new LES tool for aerosol-cloud interactions research

- ▶ novel cloud/aerosol microphysics models,



- ▶ CCN activation
- ▶ condensational growth



- ▶ collisional growth
- ▶ aqueous chemistry



- ▶ precipitation
- ▶ wet deposition
- ▶ droplet deactivation

# project architects & goals

Hanna Pawłowska, Piotr Smolarkiewicz & Wojtek Grabowski

develop a new LES tool for aerosol-cloud interactions research

- ▶ novel cloud/aerosol microphysics models,
- ▶ state-of-the-art numerical schemes,



- ▶ CCN activation
- ▶ condensational growth



- ▶ collisional growth
- ▶ aqueous chemistry



- ▶ precipitation
- ▶ wet deposition
- ▶ droplet deactivation

# project architects & goals

Hanna Pawłowska, Piotr Smolarkiewicz & Wojtek Grabowski

develop a new LES tool for aerosol-cloud interactions research

- ▶ novel cloud/aerosol microphysics models,
  - ▶ state-of-the-art numerical schemes,
  - ▶ modern coding techniques
- ↪ priorities: researchers' productivity & result reproducibility



- ▶ CCN activation
- ▶ condensational growth



- ▶ collisional growth
- ▶ aqueous chemistry



- ▶ precipitation
- ▶ wet deposition
- ▶ droplet deactivation



# Plan of the talk

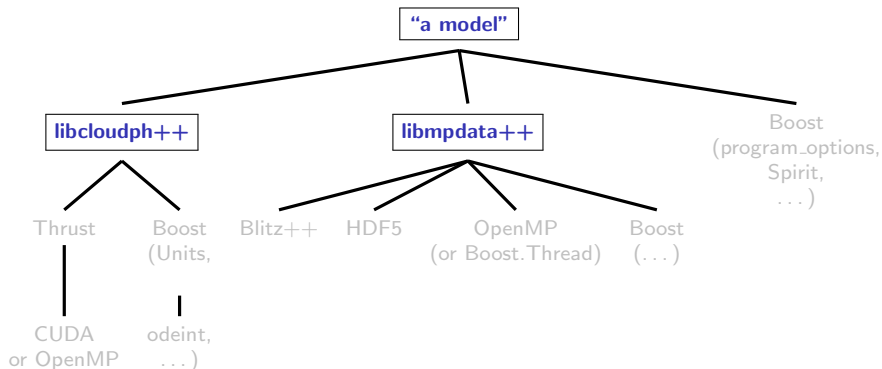
- design choices and their rationale
- example results with 2D kinematic set-up
- validation of libcloudph++'s Super-Droplet scheme
- summary & future plans

# Plan of the talk

- design choices and their rationale
- example results with 2D kinematic set-up
- validation of libcloudph++'s Super-Droplet scheme
- summary & future plans

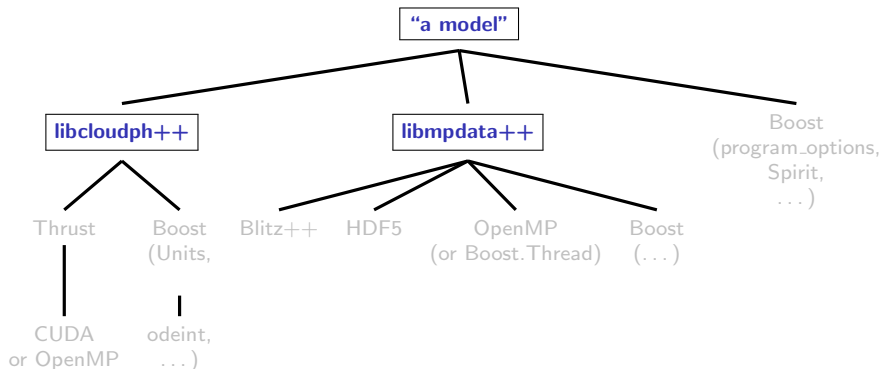
## a few words on first design choices

- ▶ structure the code into “standalone” libraries



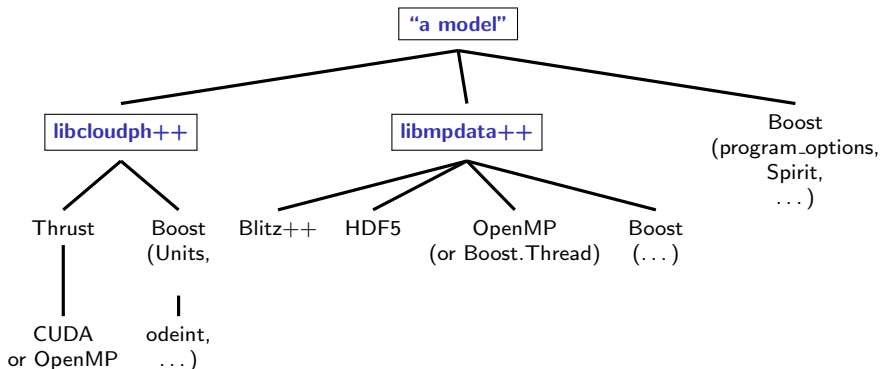
## a few words on first design choices

- ▶ structure the code into “standalone” libraries
  - ↪ easier to document, to test and to contribute to
  - ↪ easier to be reused by others (in various contexts)



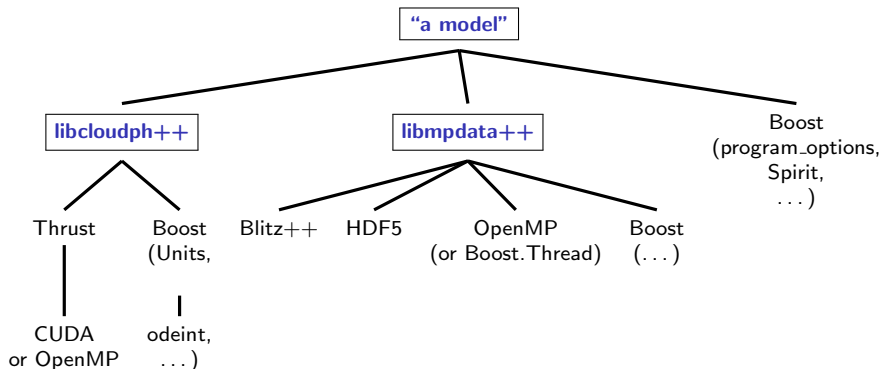
## a few words on first design choices

- ▶ structure the code into “standalone” libraries
  - ↪ easier to document, to test and to contribute to
  - ↪ easier to be reused by others (in various contexts)
- ▶ leverage existing **reusable** software



## a few words on first design choices

- ▶ structure the code into “standalone” libraries
  - ↪ easier to document, to test and to contribute to
  - ↪ easier to be reused by others (in various contexts)
- ▶ leverage existing **reusable** software
  - ↪ save time, benefit from state-of-the-art components



# libmpdata++ & libcloudph++

## project target

LES-type tool featuring:

- ▶ robust numerics (MPDATA)
- ▶ particle-based aerosol/warm-rain  $\mu$ -physics (super-droplet)

# libmpdata++ & libcloudph++

## project target

LES-type tool featuring:

- ▶ robust numerics (MPDATA)
- ▶ particle-based aerosol/warm-rain  $\mu$ -physics (super-droplet)

## current “products” – C++ libraries

**libmpdata++** parallel solvers for systems of transport equations

- ▶ <http://libmpdataxx.igf.fuw.edu.pl/>
- ▶ [doi:10.5194/gmd-8-1005-2015](https://doi.org/10.5194/gmd-8-1005-2015)

**libcloudph++** aerosol/cloud  $\mu$ -physics algorithm collection

- ▶ <http://libcloudphxx.igf.fuw.edu.pl/>
- ▶ [doi:10.5194/gmdd-7-8275-2014](https://doi.org/10.5194/gmdd-7-8275-2014)



## libcloudph++ components

- ▶ single-moment bulk saturation-adjustment scheme with Kessler's coalescence
- ▶ double-moment bulk scheme with predicted supersaturation (Morrison & Grabowski 2007)
- ▶ particle-based scheme with Monte-Carlo coalescence (super-droplet method of Shima et al. 2009)
- ▶ ...

## libcloudph++ components

- ▶ single-moment bulk saturation-adjustment scheme with Kessler's coalescence
- ▶ double-moment bulk scheme with predicted supersaturation (Morrison & Grabowski 2007)
- ▶ particle-based scheme with Monte-Carlo coalescence (super-droplet method of Shima et al. 2009)
- ▶ ...



- ▶ CCN activation
- ▶ condensational growth

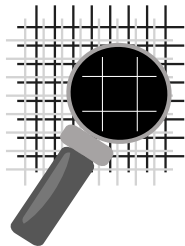


- ▶ collisional growth
- ▶ aqueous chemistry

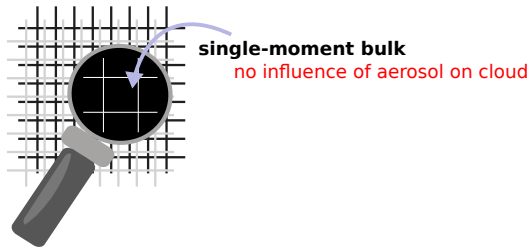


- ▶ precipitation
- ▶ wet deposition
- ▶ droplet deactivation

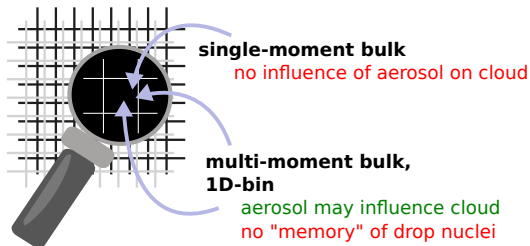
## Eulerian vs. Lagrangian $\mu$ -physics



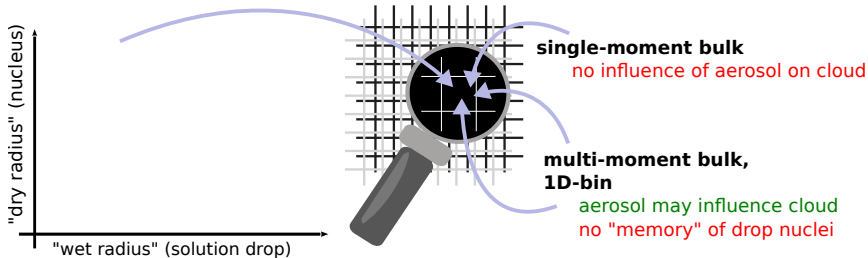
# Eulerian vs. Lagrangian $\mu$ -physics



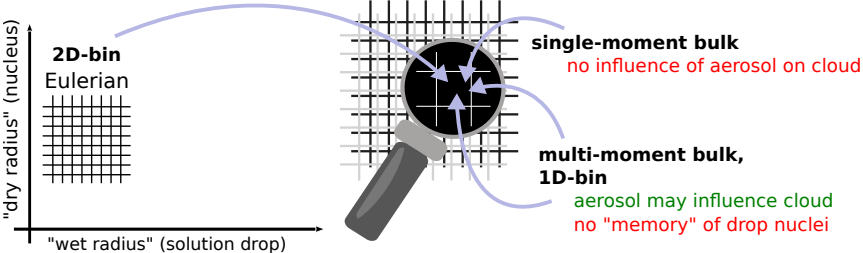
# Eulerian vs. Lagrangian $\mu$ -physics



# Eulerian vs. Lagrangian $\mu$ -physics

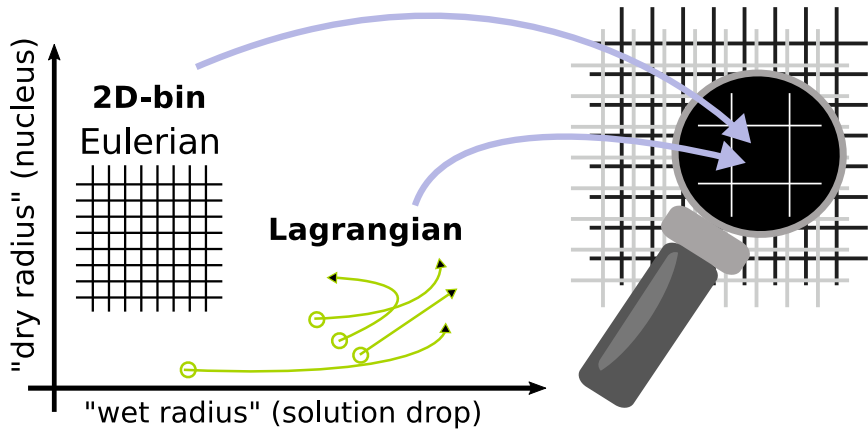


# Eulerian vs. Lagrangian $\mu$ -physics





# Eulerian vs. Lagrangian $\mu$ -physics



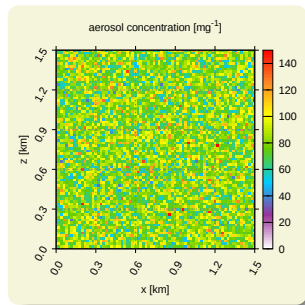
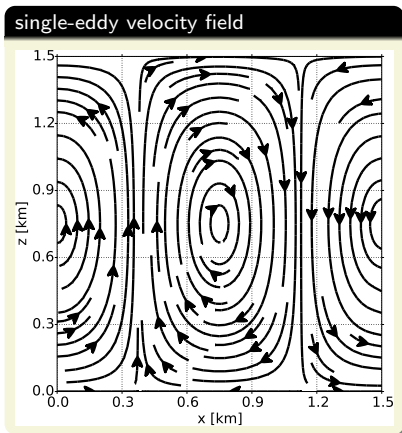
# Plan of the talk

- design choices and their rationale
- example results with 2D kinematic set-up
- validation of libcloudph++'s Super-Droplet scheme
- summary & future plans

# Plan of the talk

- design choices and their rationale
- example results with 2D kinematic set-up
- validation of libcloudph++'s Super-Droplet scheme
- summary & future plans

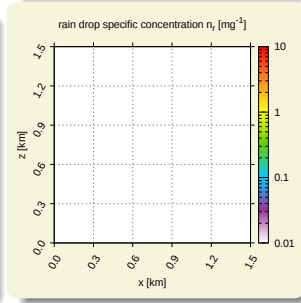
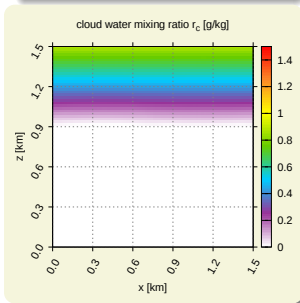
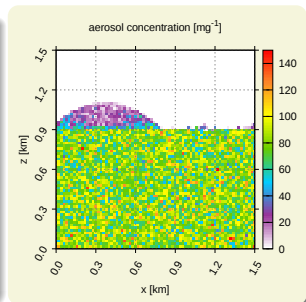
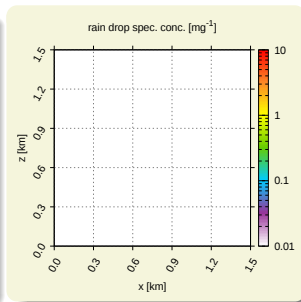
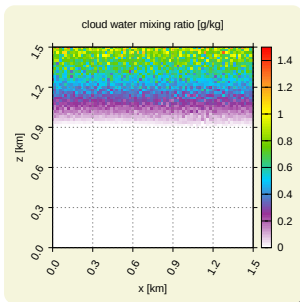
# libcloudph++: VOCALS-inspired aerosol processing set-up



- ▶ set-up: Grabowski & Lebo (ICMW 2012)
- ▶ 2D prescribed flow
- ▶ advection: `libmpdata++` (2-pass FCT)
- ▶  $\mu$ -physics: `libcloudph++`

# libcloudph++: VOCALS-inspired aerosol processing set-up

x



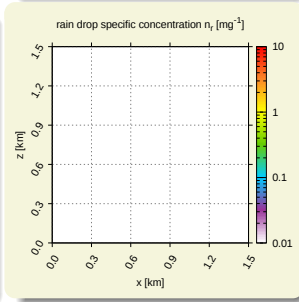
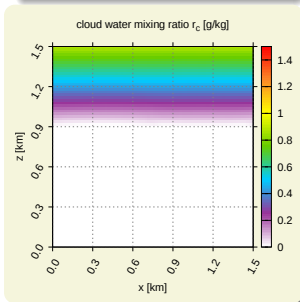
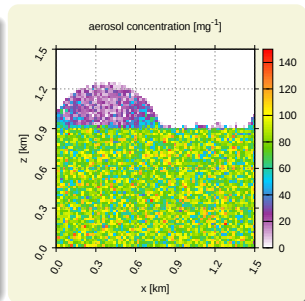
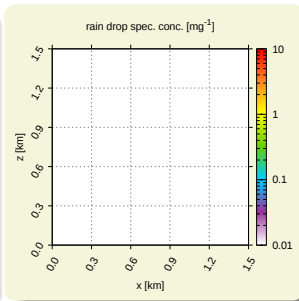
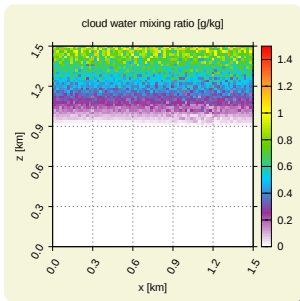
Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

Two arrows point from the text to the plots: one from the Lagrangian/Monte-Carlo scheme to the aerosol concentration plot, and one from the 2-moment bulk scheme to the rain drop specific concentration plot.

# libcloudph++: VOCALS-inspired aerosol processing set-up

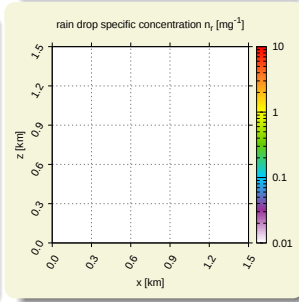
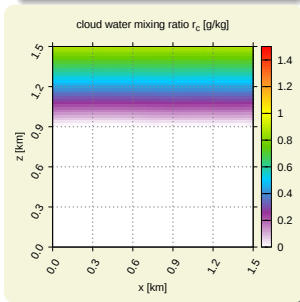
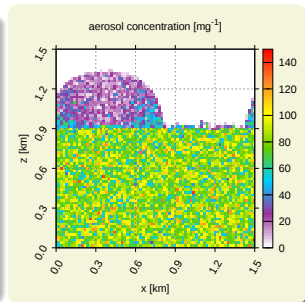
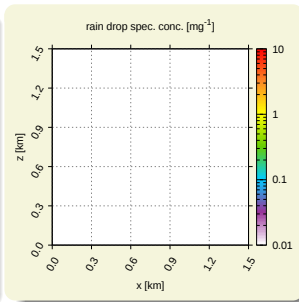
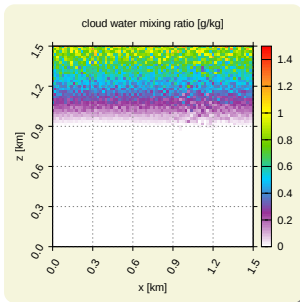
xx



Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

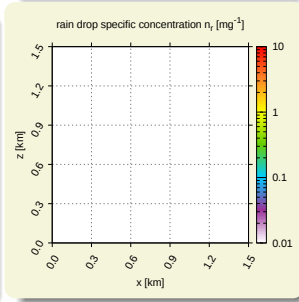
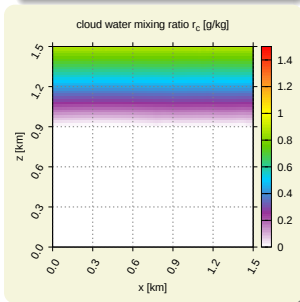
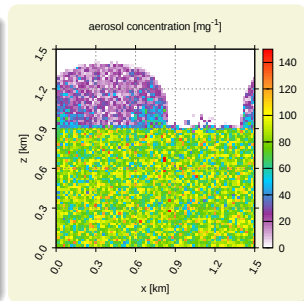
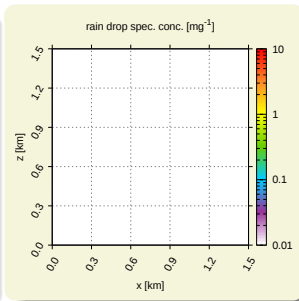
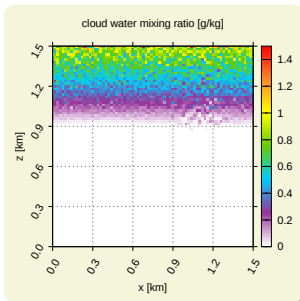
# libcloudph++: VOCALS-inspired aerosol processing set-up xxx



Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

# libcloudph++: VOCALS-inspired aerosol processing set-up XXXX

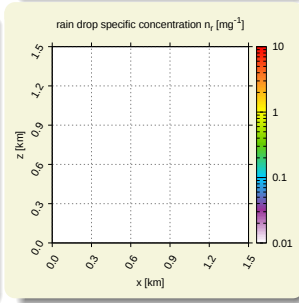
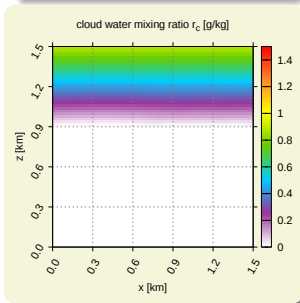
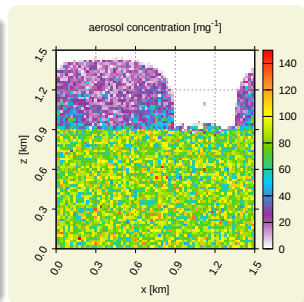
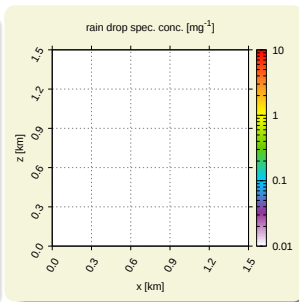
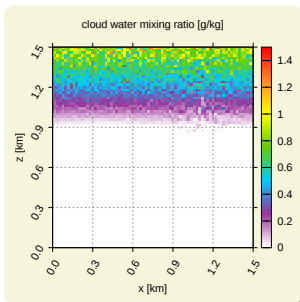


Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)



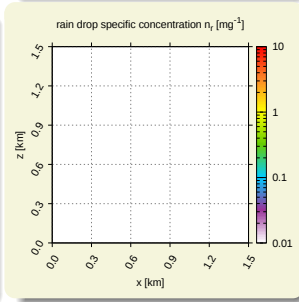
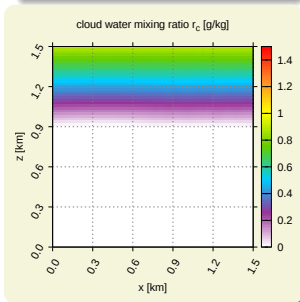
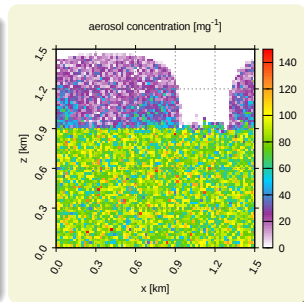
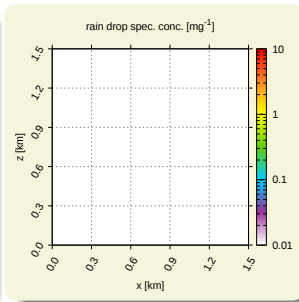
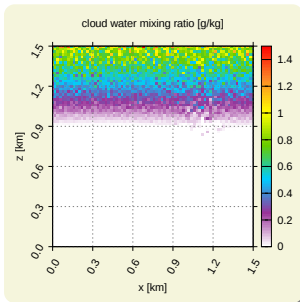
# libcloudph++: VOCALS-inspired aerosol processing set-up XXXXX



Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

# libcloudph++: VOCALS-inspired aerosol processing set-up XXXXXX

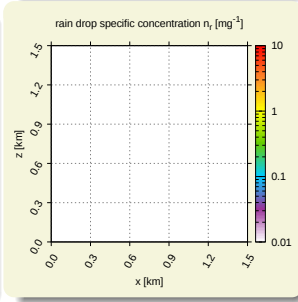
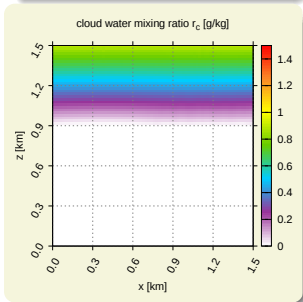
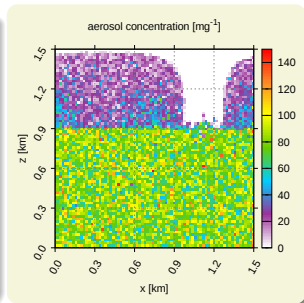
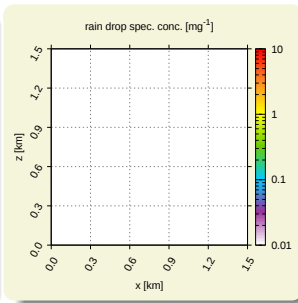
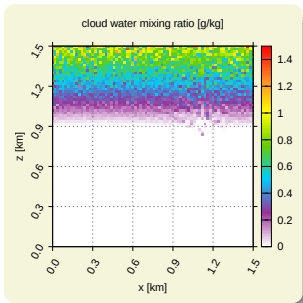


Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

# libcloudph++: VOCALS-inspired aerosol processing set-up

XXXXXXXX

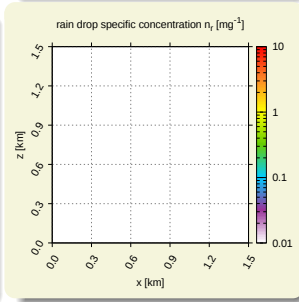
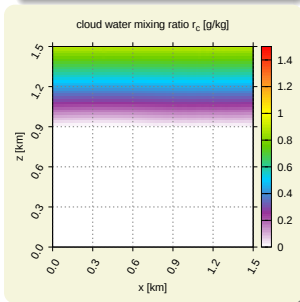
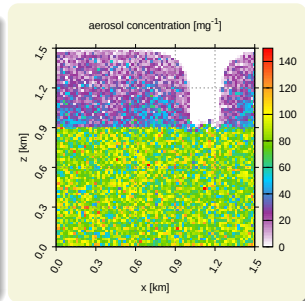
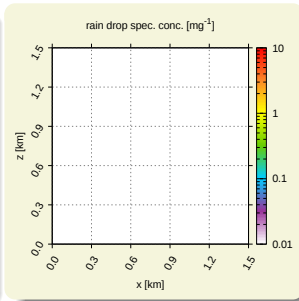
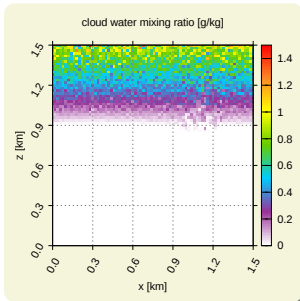


Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

# libcloudph++: VOCALS-inspired aerosol processing set-up

XXXXXXXXXX

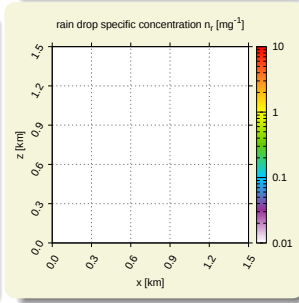
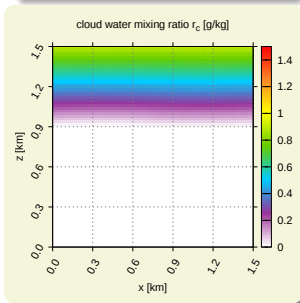
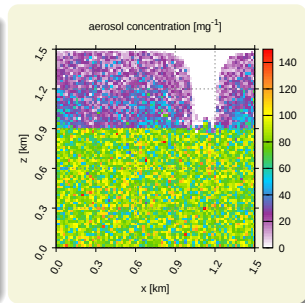
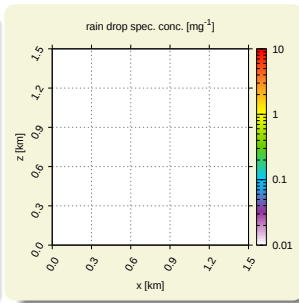
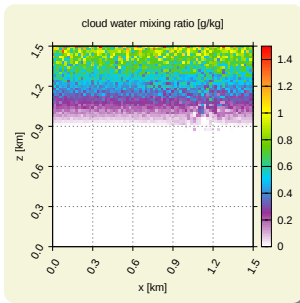


Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

# libcloudph++: VOCALS-inspired aerosol processing set-up

XXXXXXXXXX

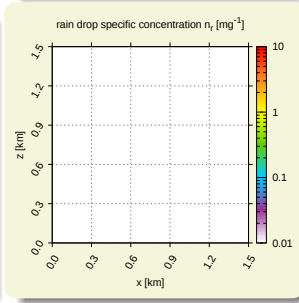
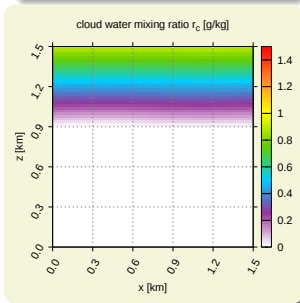
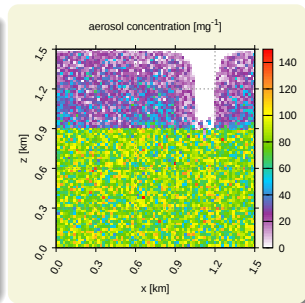
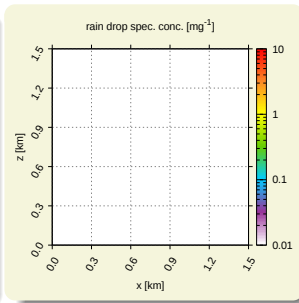
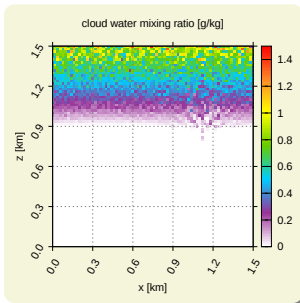


Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

# libcloudph++: VOCALS-inspired aerosol processing set-up

XXXXXXXXXXXX

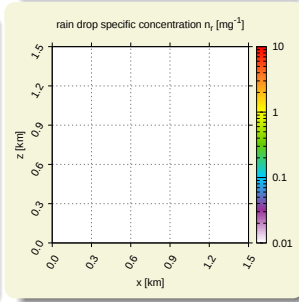
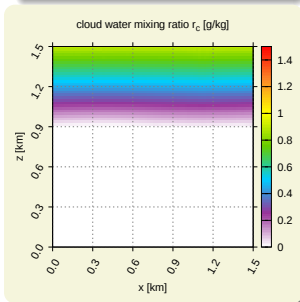
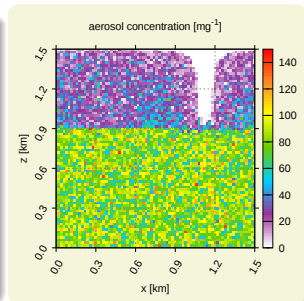
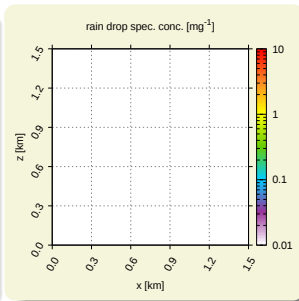
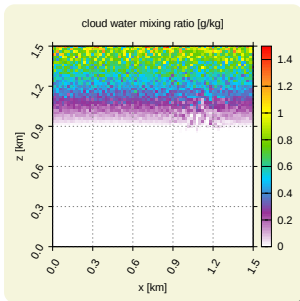


Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

# libcloudph++: VOCALS-inspired aerosol processing set-up

XXXXXXXXXXXX

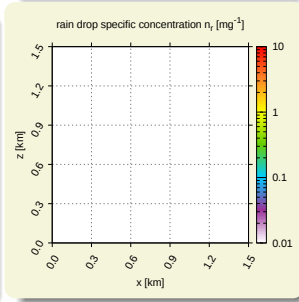
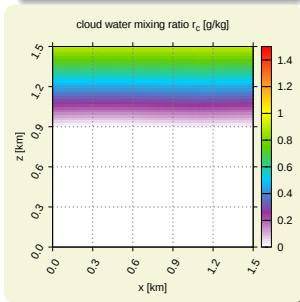
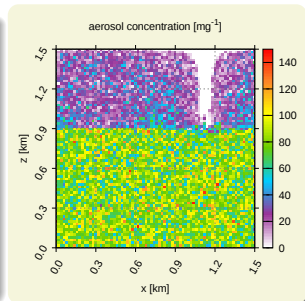
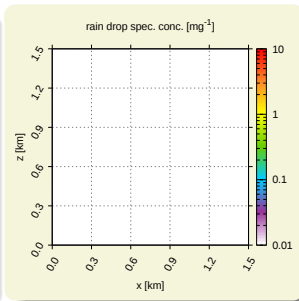
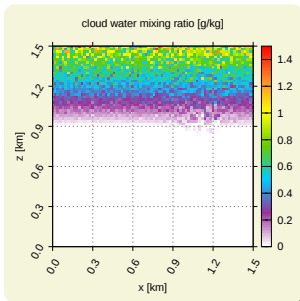


Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

# libcloudph++: VOCALS-inspired aerosol processing set-up

XXXXXXXXXXXXXX



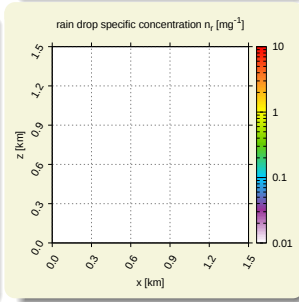
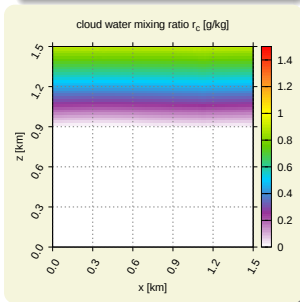
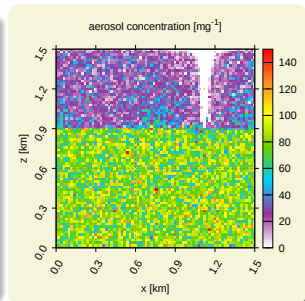
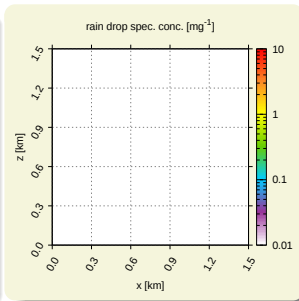
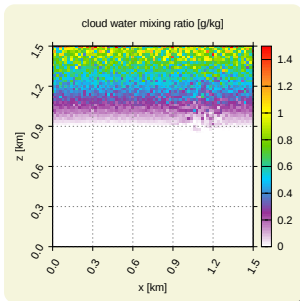
Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)



# libcloudph++: VOCALS-inspired aerosol processing set-up

XXXXXXXXXXXXXXXXXX

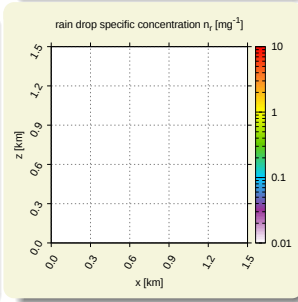
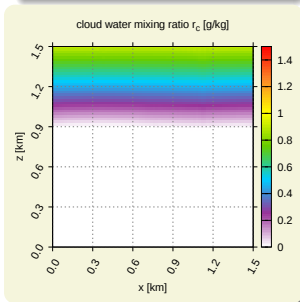
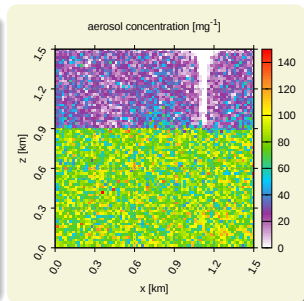
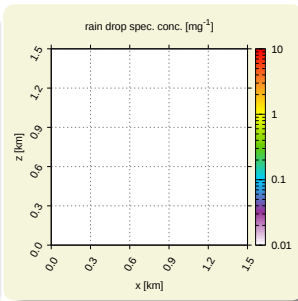
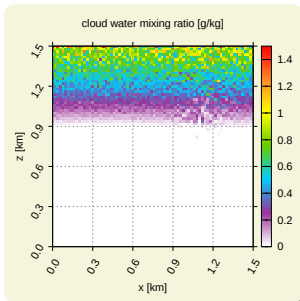


Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

# libcloudph++: VOCALS-inspired aerosol processing set-up

XXXXXXXXXXXXXXXXXX

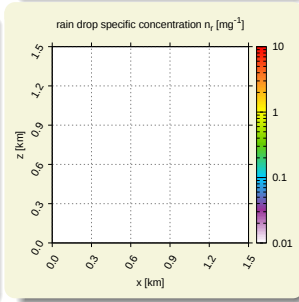
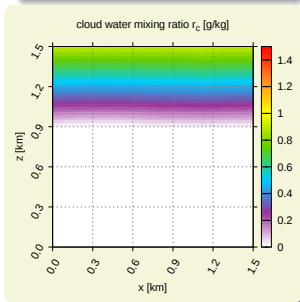
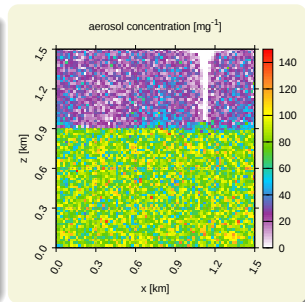
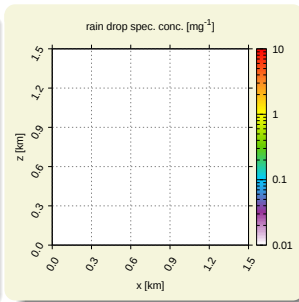
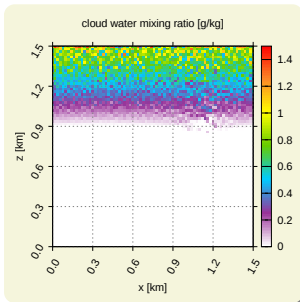


Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

# libcloudph++: VOCALS-inspired aerosol processing set-up

XXXXXXXXXXXXXXXXXX

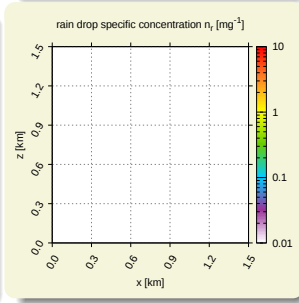
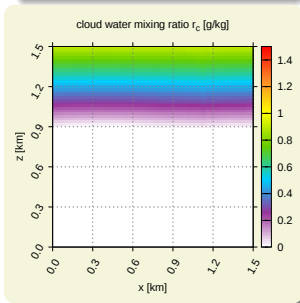
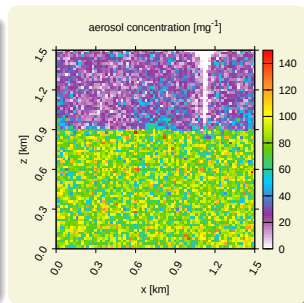
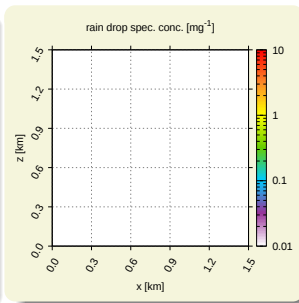
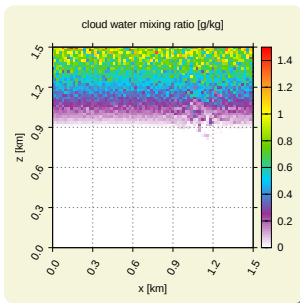


Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

# libcloudph++: VOCALS-inspired aerosol processing set-up

XXXXXXXXXXXXXXXXXXXX

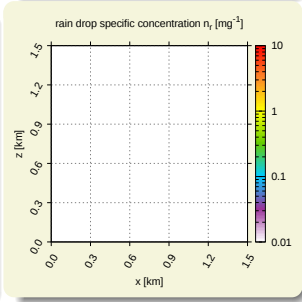
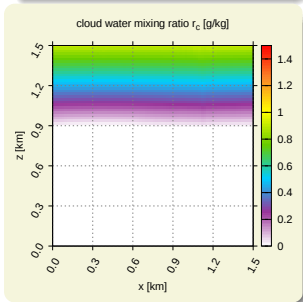
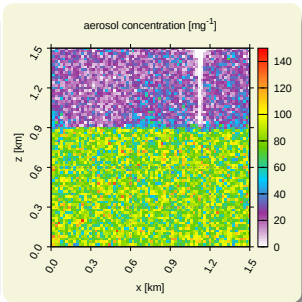
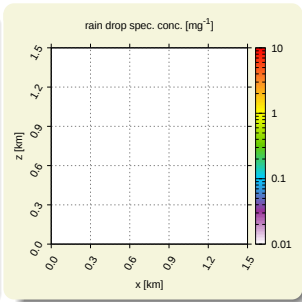
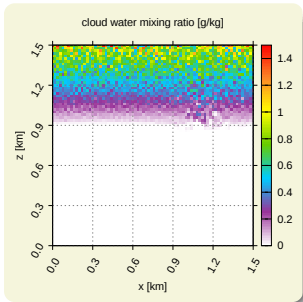


Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

# libcloudph++: VOCALS-inspired aerosol processing set-up

XXXXXXXXXXXXXXXXXXXX

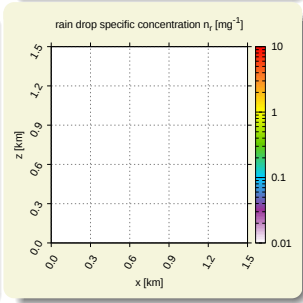
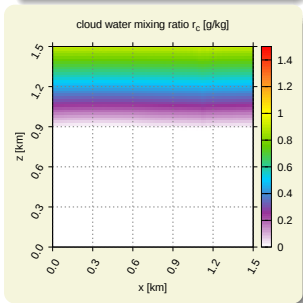
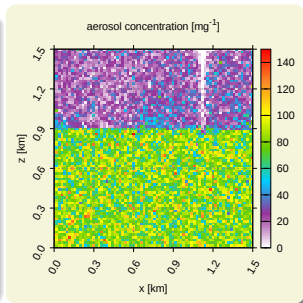
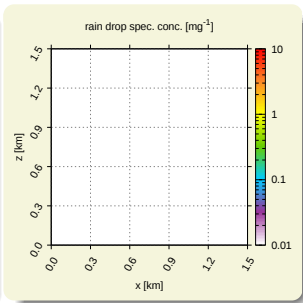
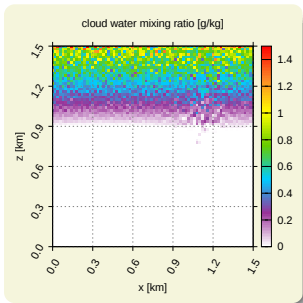


Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

# libcloudph++: VOCALS-inspired aerosol processing set-up

XXXXXXXXXXXXXXXXXXXX

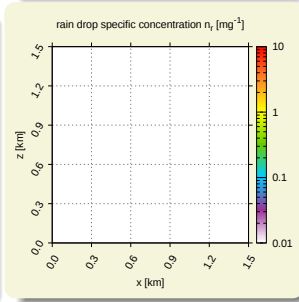
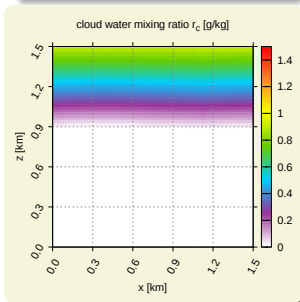
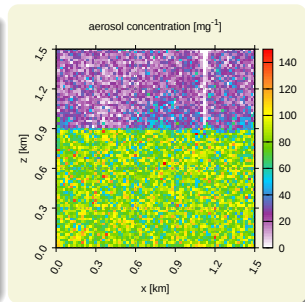
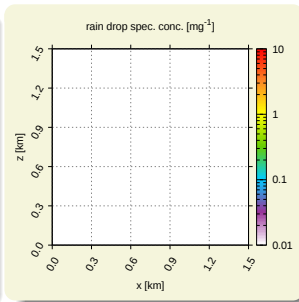
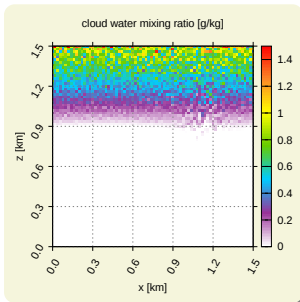


Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

# libcloudph++: VOCALS-inspired aerosol processing set-up

XXXXXXXXXXXXXXXXXXXXXXXXXX



↖

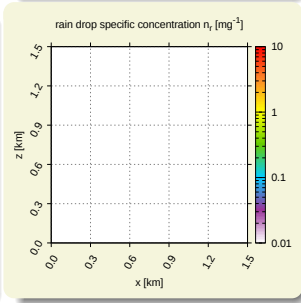
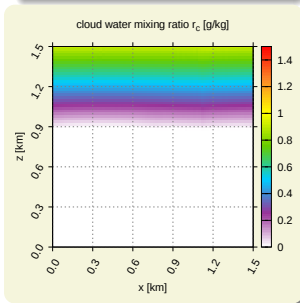
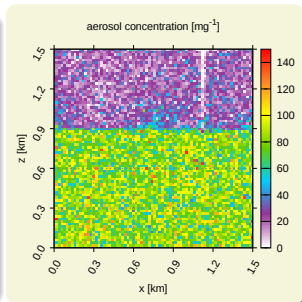
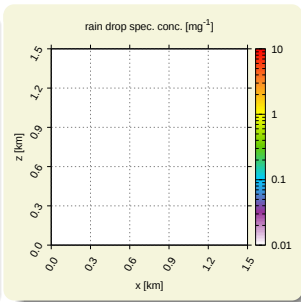
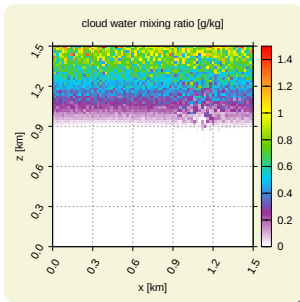
Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

←

# libcloudph++: VOCALS-inspired aerosol processing set-up

XXXXXXXXXXXXXXXXXXXXXXXXXX



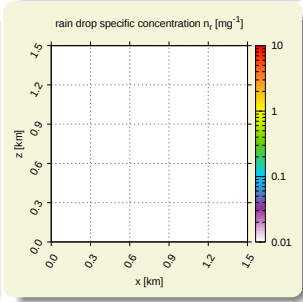
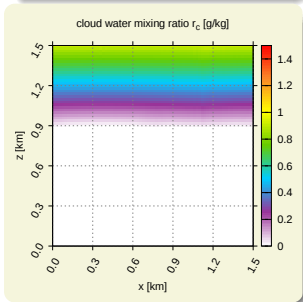
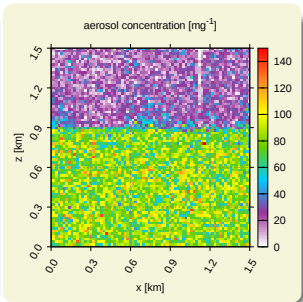
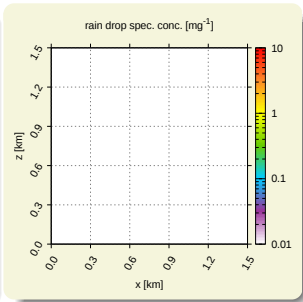
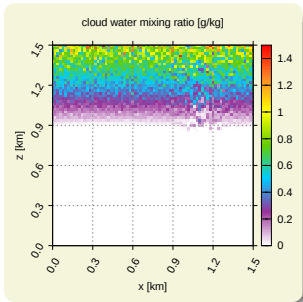
Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)



# libcloudph++: VOCALS-inspired aerosol processing set-up

XXXXXXXXXXXXXXXXXXXXXXXXXXXX

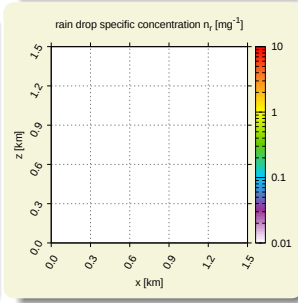
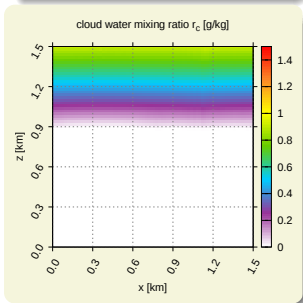
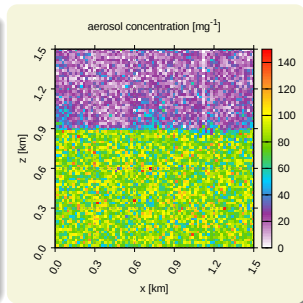
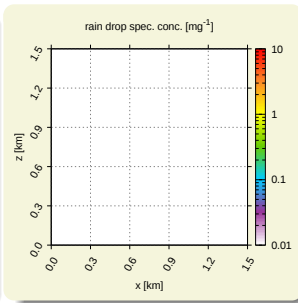
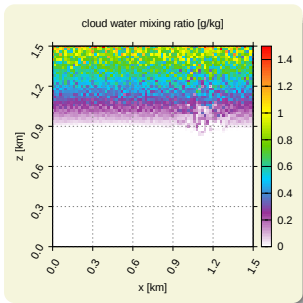


Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

# libcloudph++: VOCALS-inspired aerosol processing set-up

XXXXXXXXXXXXXXXXXXXXXXXXXXXX

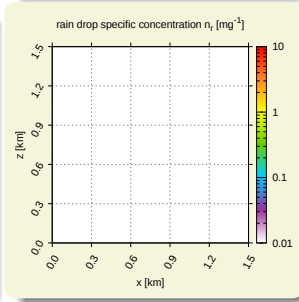
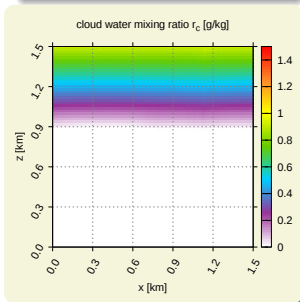
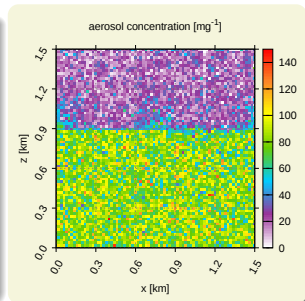
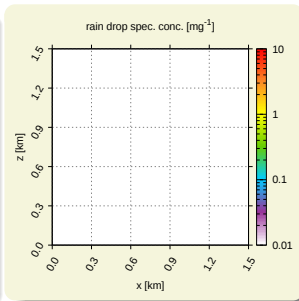
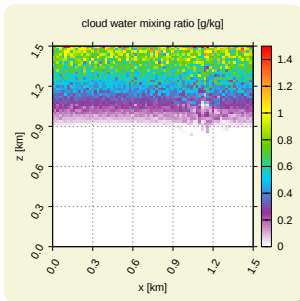


Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

# libcloudph++: VOCALS-inspired aerosol processing set-up

XXXXXXXXXXXXXXXXXXXXXXXXXXXX

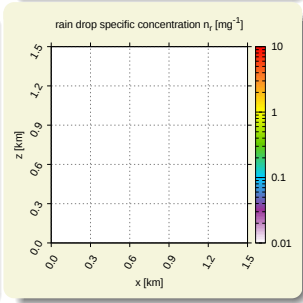
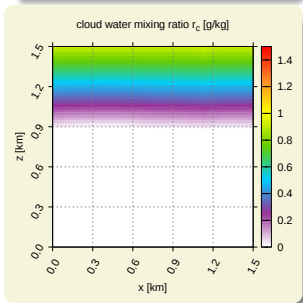
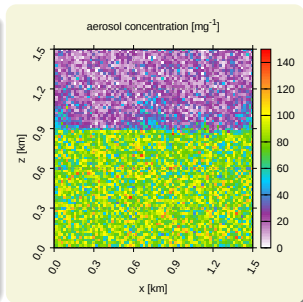
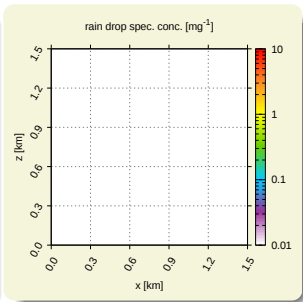
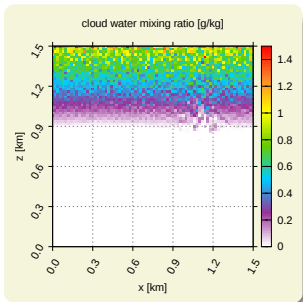


Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

# libcloudph++: VOCALS-inspired aerosol processing set-up

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

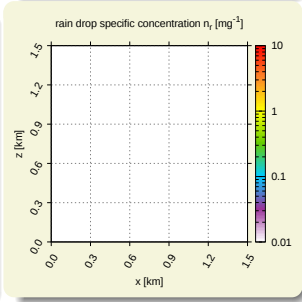
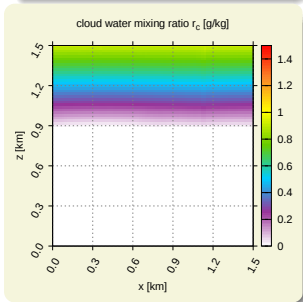
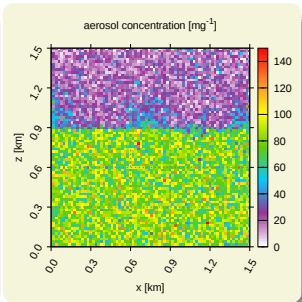
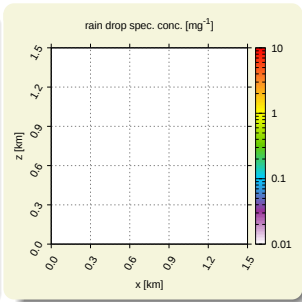
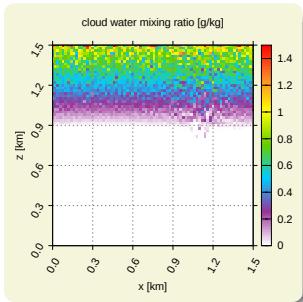


Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

# libcloudph++: VOCALS-inspired aerosol processing set-up

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

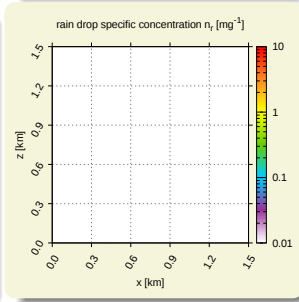
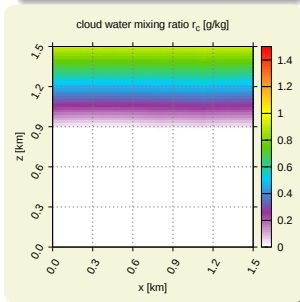
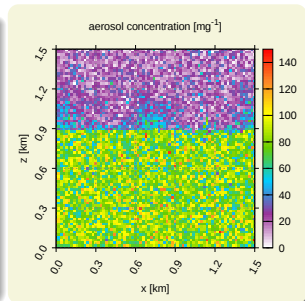
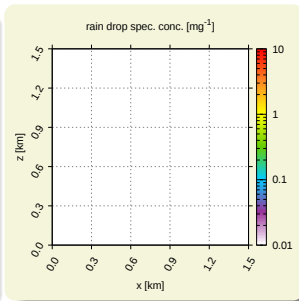
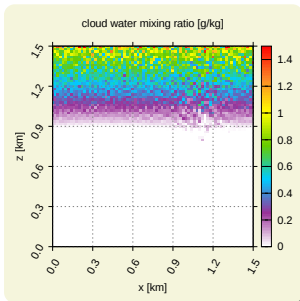


Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

# libcloudph++: VOCALS-inspired aerosol processing set-up

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

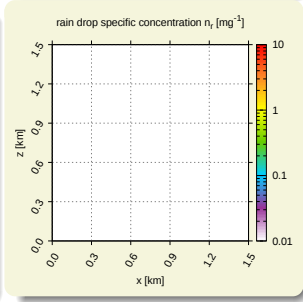
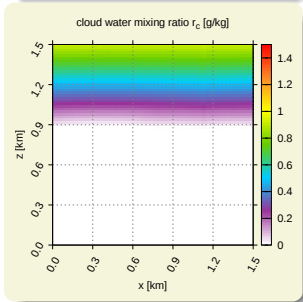
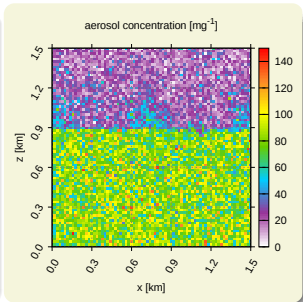
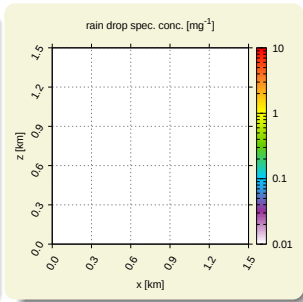
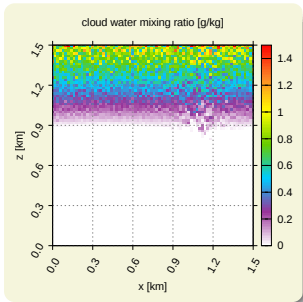


Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

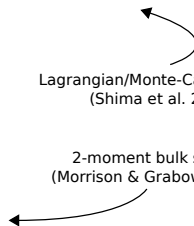
# libcloudph++: VOCALS-inspired aerosol processing set-up

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX



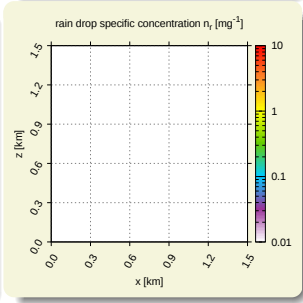
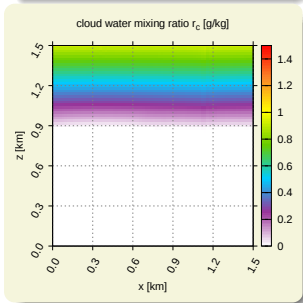
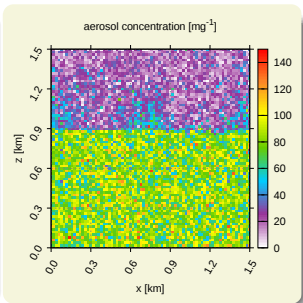
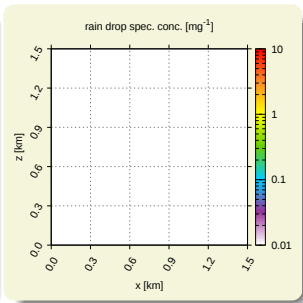
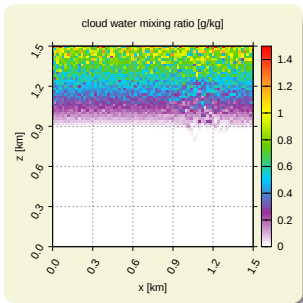
Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)



# libcloudph++: VOCALS-inspired aerosol processing set-up

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX



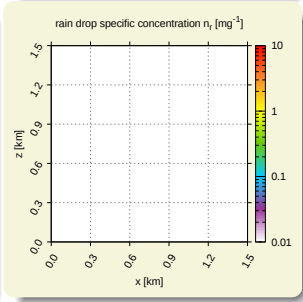
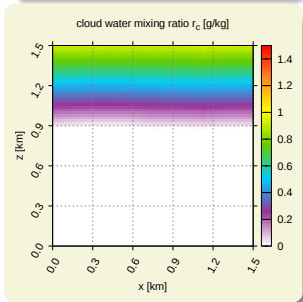
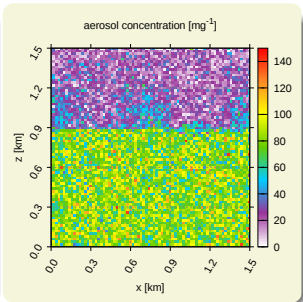
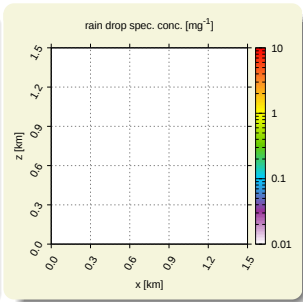
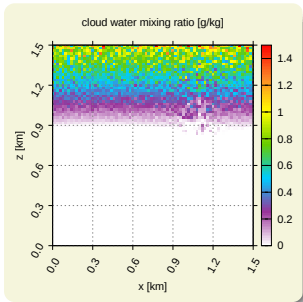
Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)



# libcloudph++: VOCALS-inspired aerosol processing set-up

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

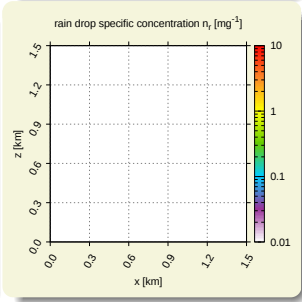
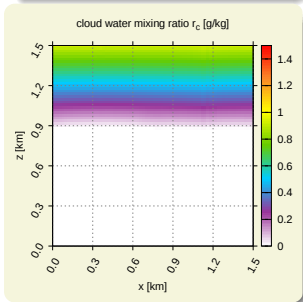
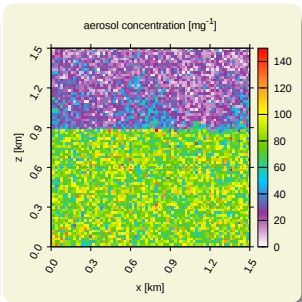
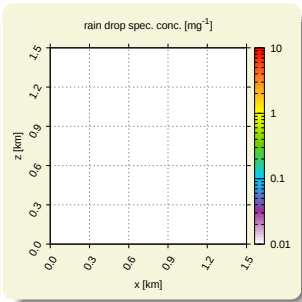
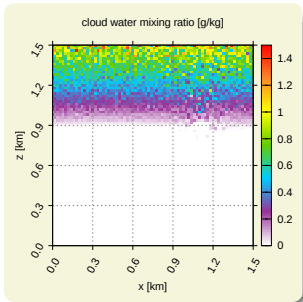


Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

# libcloudph++: VOCALS-inspired aerosol processing set-up

XX

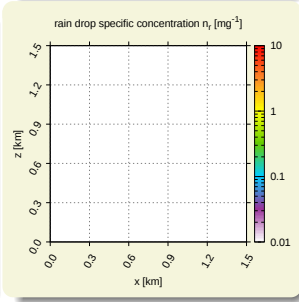
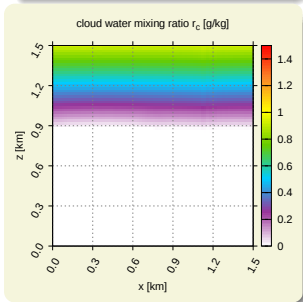
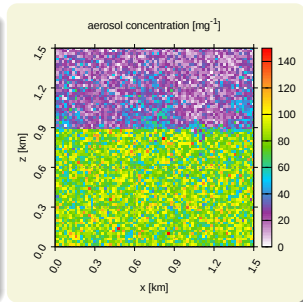
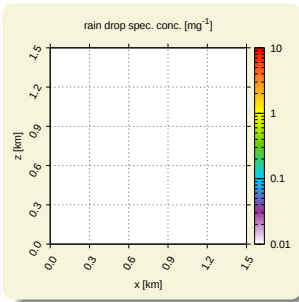
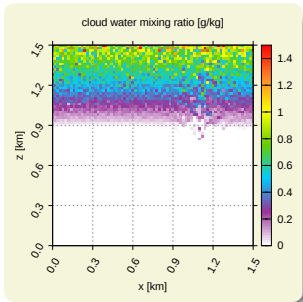


Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

# libcloudph++: VOCALS-inspired aerosol processing set-up

XX



↖

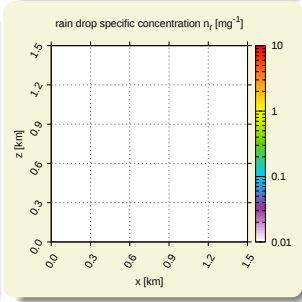
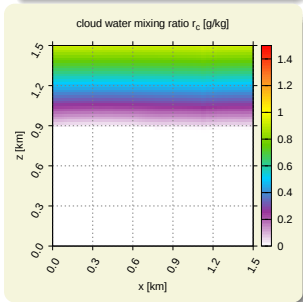
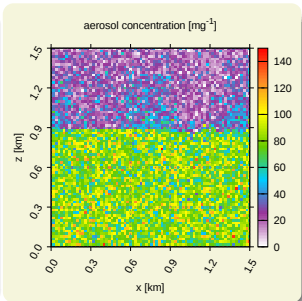
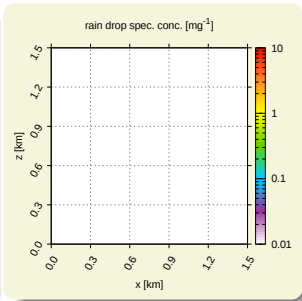
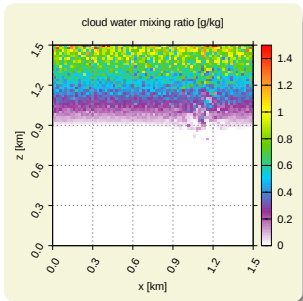
Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

←

# libcloudph++: VOCALS-inspired aerosol processing set-up

XX



↖

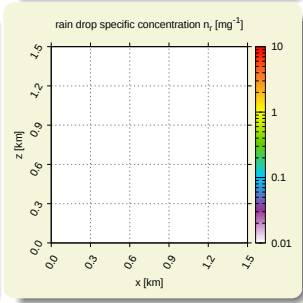
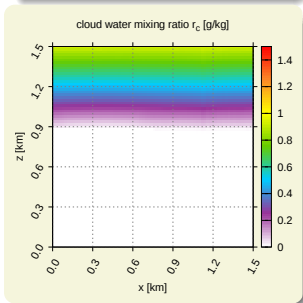
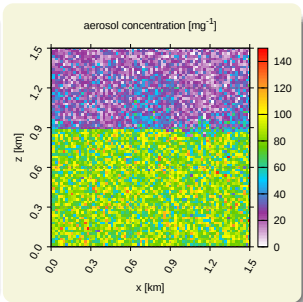
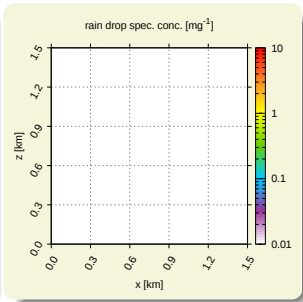
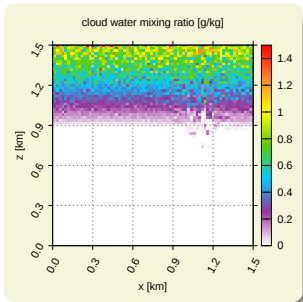
Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

↙

# libcloudph++: VOCALS-inspired aerosol processing set-up

XX

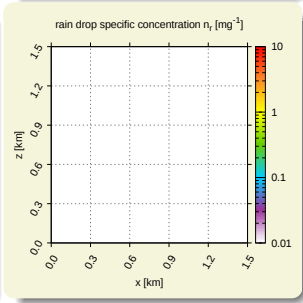
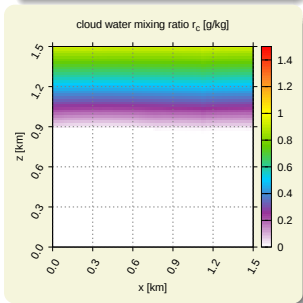
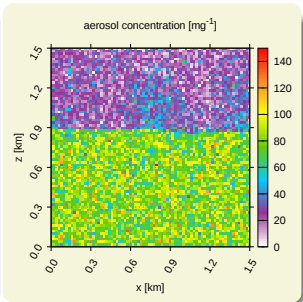
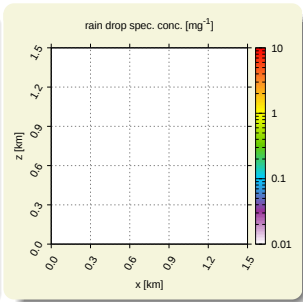
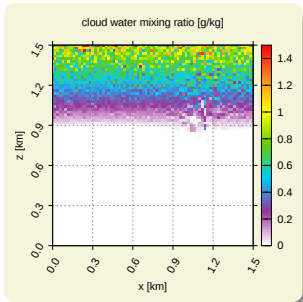


Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

# libcloudph++: VOCALS-inspired aerosol processing set-up

XX

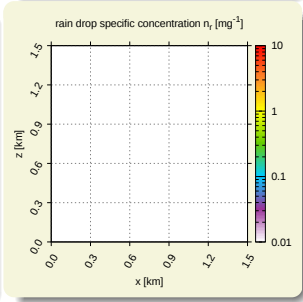
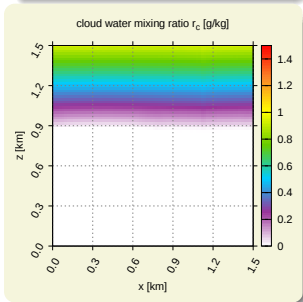
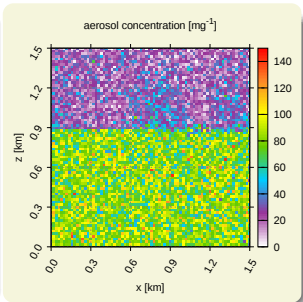
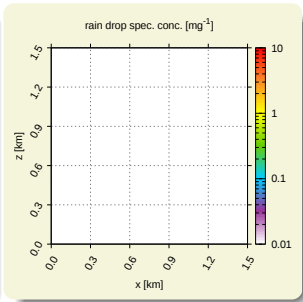
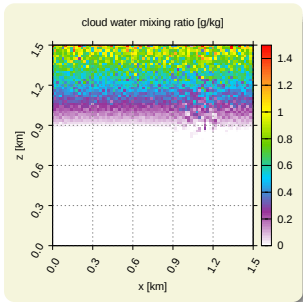


Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

# libcloudph++: VOCALS-inspired aerosol processing set-up

XX

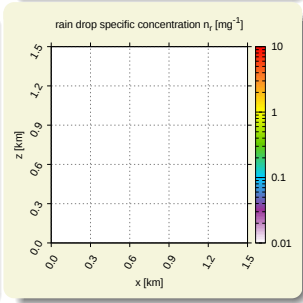
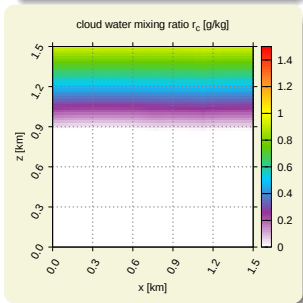
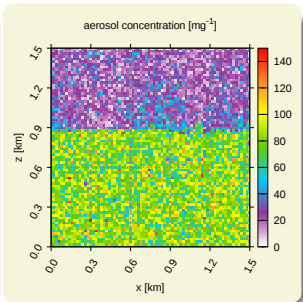
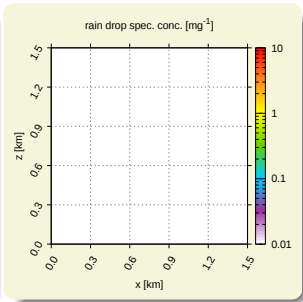
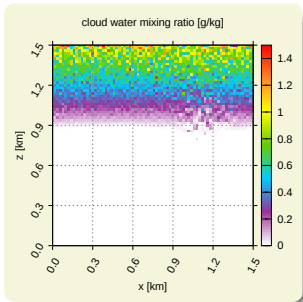


Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

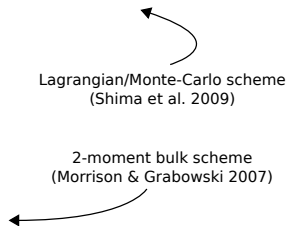
# libcloudph++: VOCALS-inspired aerosol processing set-up

XX



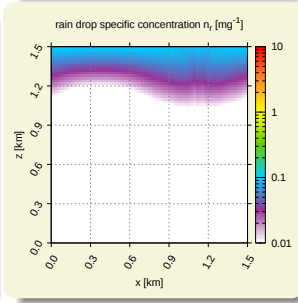
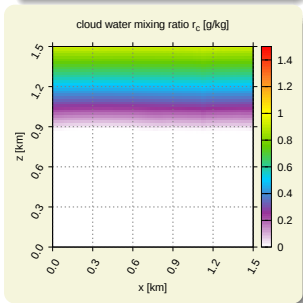
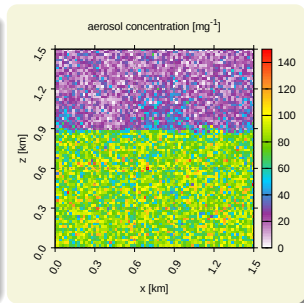
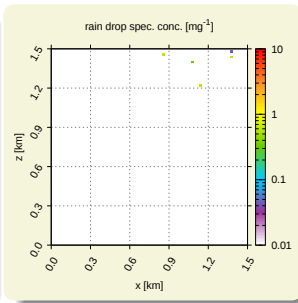
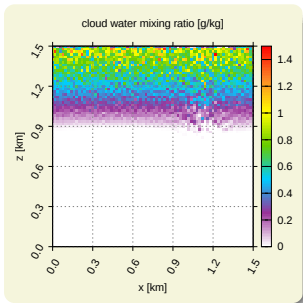
Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)





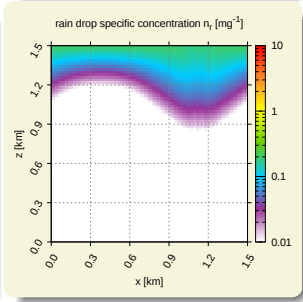
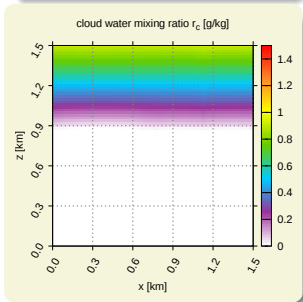
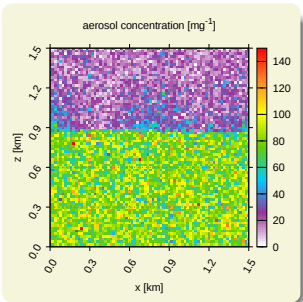
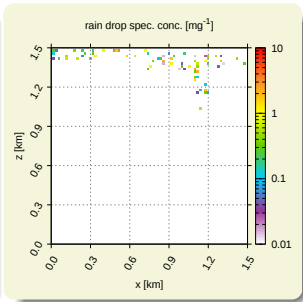
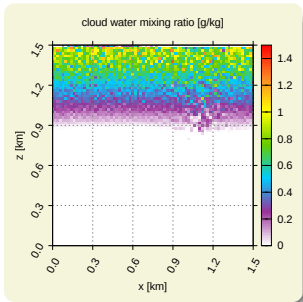
# libcloudph++: VOCALS-inspired aerosol processing set-up



Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

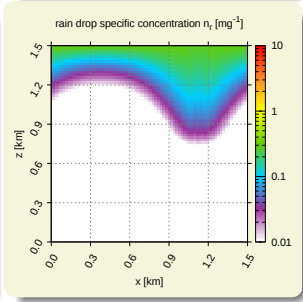
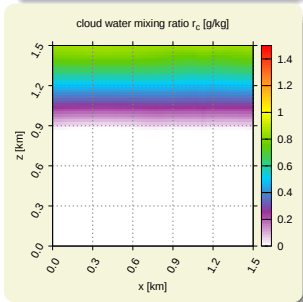
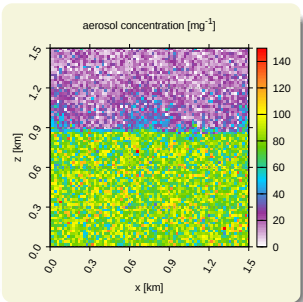
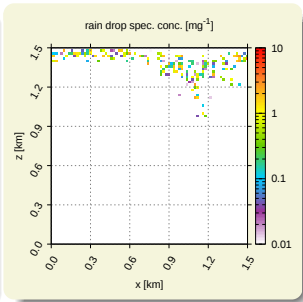
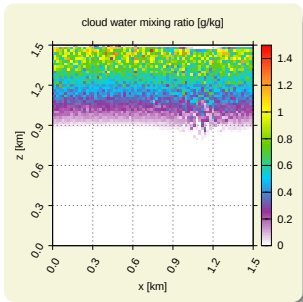
# libcloudph++: VOCALS-inspired aerosol processing set-up



Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

# libcloudph++: VOCALS-inspired aerosol processing set-up

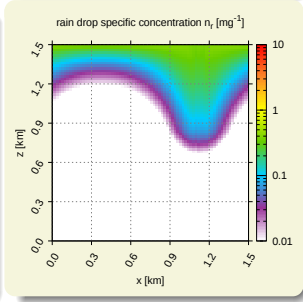
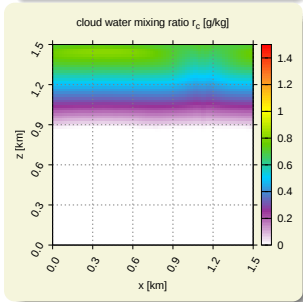
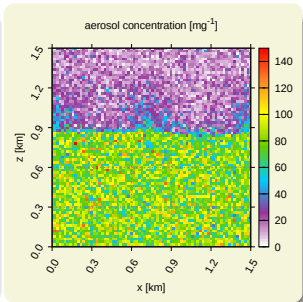
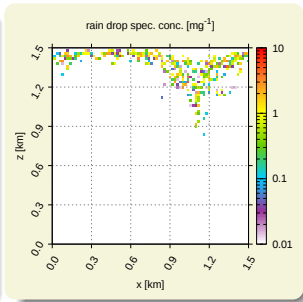
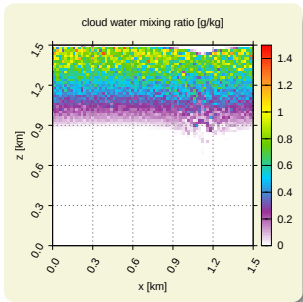


Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

# libcloudph++: VOCALS-inspired aerosol processing set-up

XXOOO

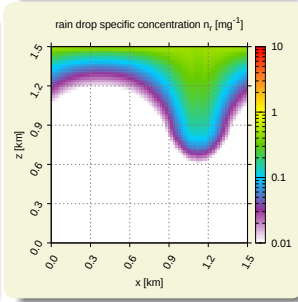
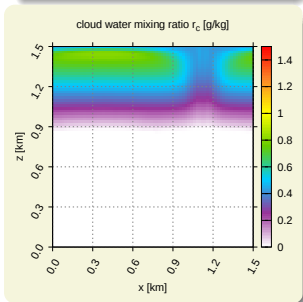
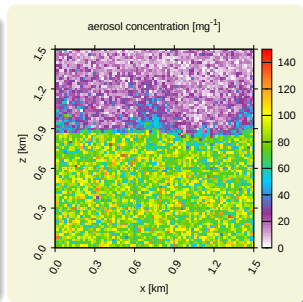
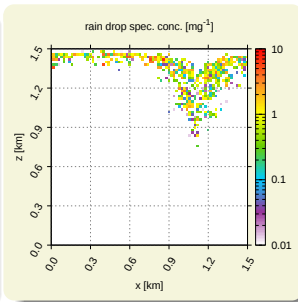
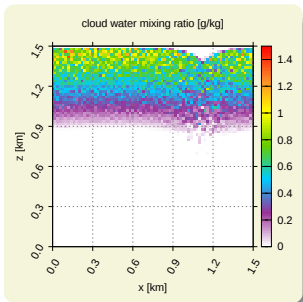


Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

# libcloudph++: VOCALS-inspired aerosol processing set-up

XXOOOO

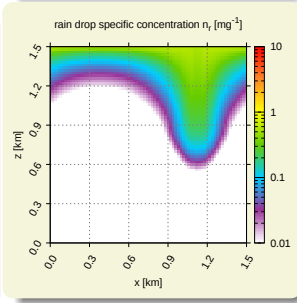
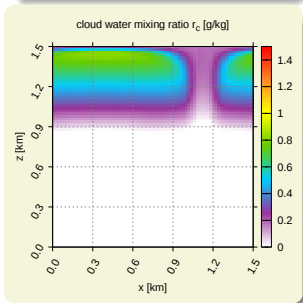
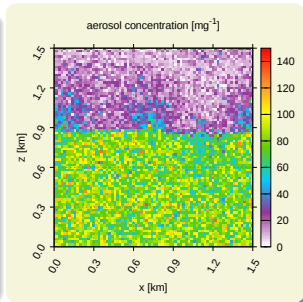
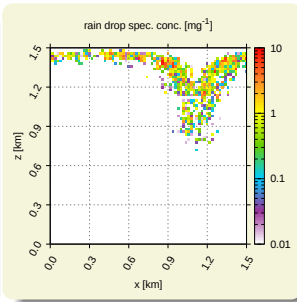
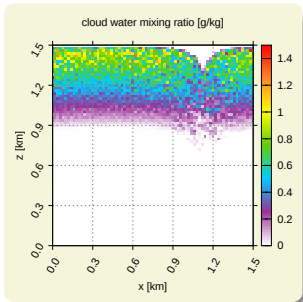


Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

# libcloudph++: VOCALS-inspired aerosol processing set-up

XXO O O O O

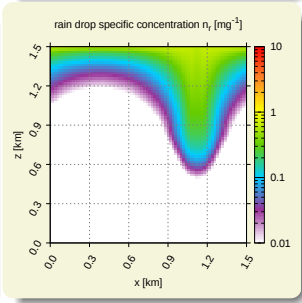
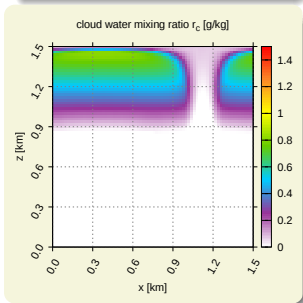
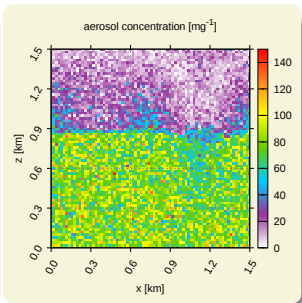
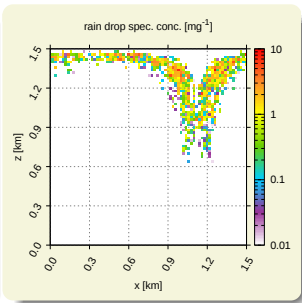
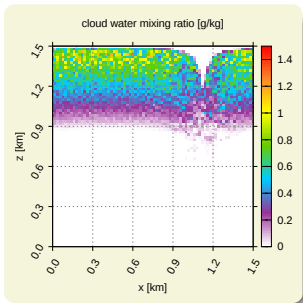


Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

# libcloudph++: VOCALS-inspired aerosol processing set-up

XXOOOOOO

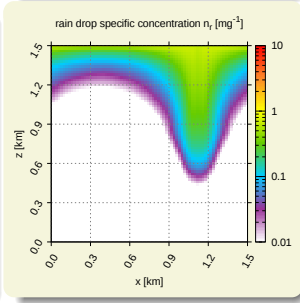
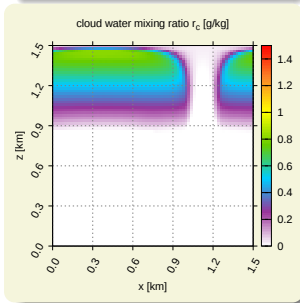
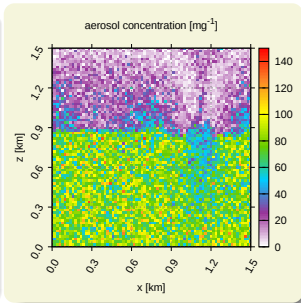
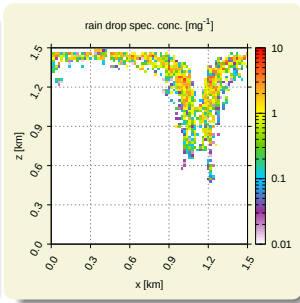
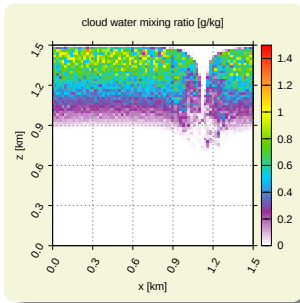


Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

# libcloudph++: VOCALS-inspired aerosol processing set-up

XX

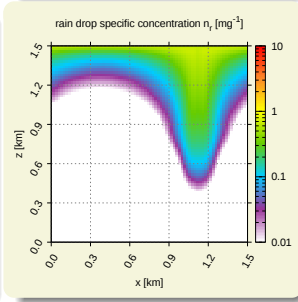
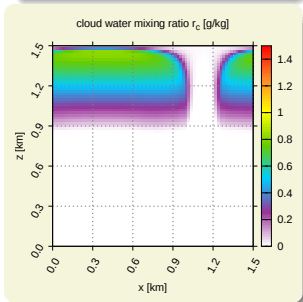
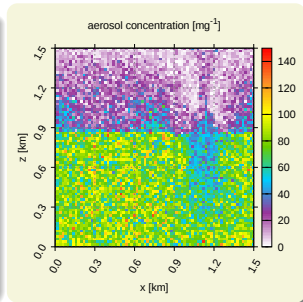
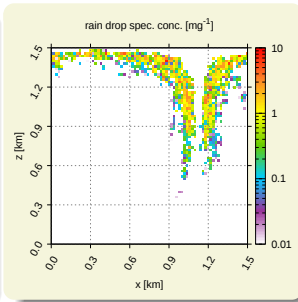
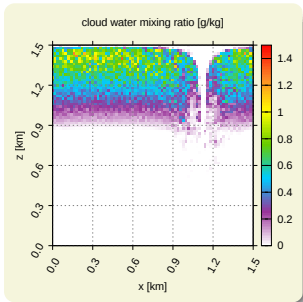


Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)



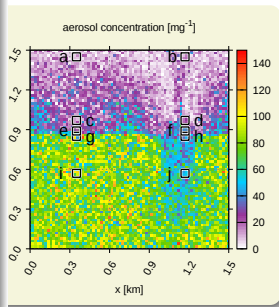
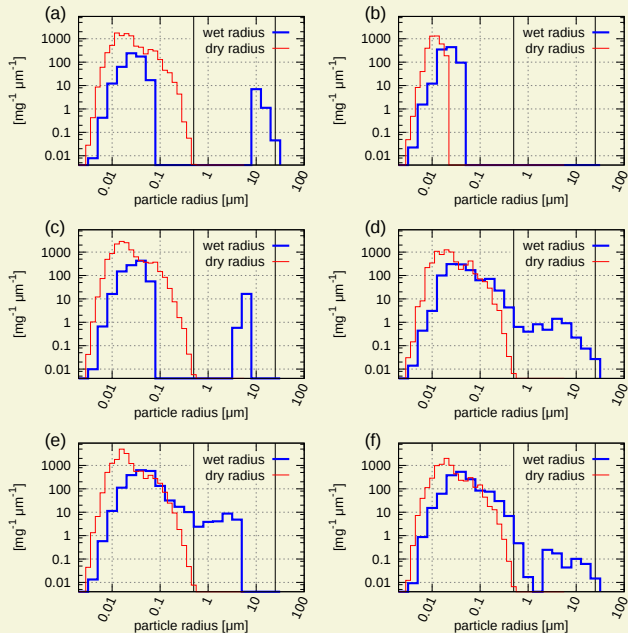
# libcloudph++: VOCALS-inspired aerosol processing set-up



Lagrangian/Monte-Carlo scheme  
(Shima et al. 2009)

2-moment bulk scheme  
(Morrison & Grabowski 2007)

## 2x2 cell particle-derived spectra



# Plan of the talk

- design choices and their rationale
- example results with 2D kinematic set-up
- validation of libcloudph++'s Super-Droplet scheme
- summary & future plans

# Plan of the talk

- design choices and their rationale
- example results with 2D kinematic set-up
- validation of libcloudph++'s Super-Droplet scheme
- summary & future plans

# Coalescence module - current status

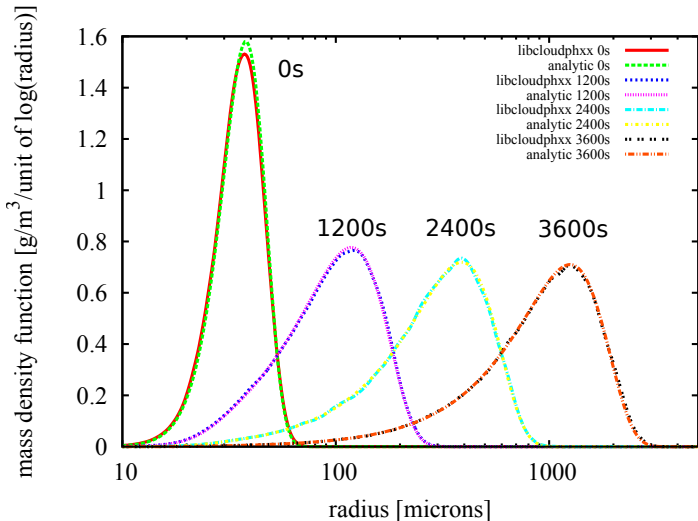
- ▶ implemented kernels
  - ▶ Golovin
  - ▶ Geometric
  - ▶ Hall
  - ▶ Hall and Davis
- ▶ terminal velocity formulas
  - ▶ Beard
  - ▶ Khvorostyanov
- ▶ execution policies
  - ▶ CPU (serial or OpenMP for multi-core CPUs)
  - ▶ GPU (CUDA for nVidia hardware)

tests following Shima et al. 2009 (section. 6)

# Coalescence module - current status

- ▶ implemented kernels
  - ▶ Golovin
  - ▶ Geometric
  - ▶ Hall
  - ▶ Hall and Davis
- ▶ terminal velocity formulas
  - ▶ Beard
  - ▶ Khvorostyanov
- ▶ execution policies
  - ▶ CPU (serial or OpenMP for multi-core CPUs)
  - ▶ GPU (CUDA for nVidia hardware)

# Coalescence validation - Golovin kernel (CPU)



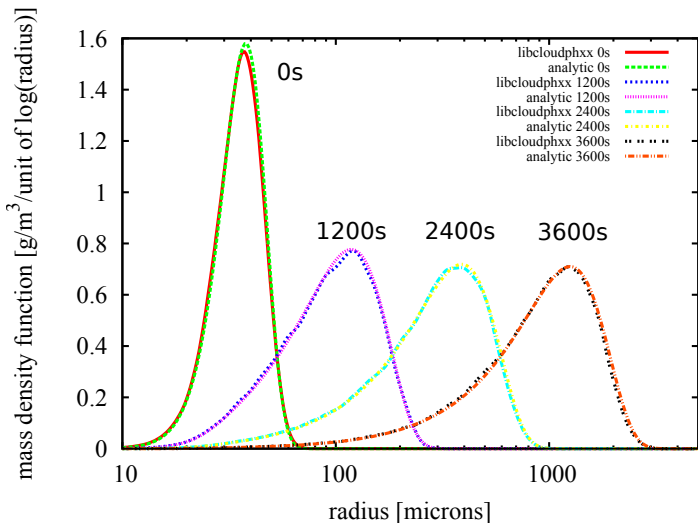
volume:  $10^6 \text{ m}^3$   
LWC:  $1 \text{ g}/\text{m}^3$   
super-droplets:  $2 \times 10^5$



# Coalescence module - current status

- ▶ implemented kernels
  - ▶ Golovin
  - ▶ Geometric
  - ▶ Hall
  - ▶ Hall and Davis
- ▶ terminal velocity formulas
  - ▶ Beard
  - ▶ Khvorostyanov
- ▶ execution policies
  - ▶ CPU (serial or OpenMP for multi-core CPUs)
  - ▶ GPU (CUDA for nVidia hardware)

# Coalescence validation - Golovin kernel (GPU)

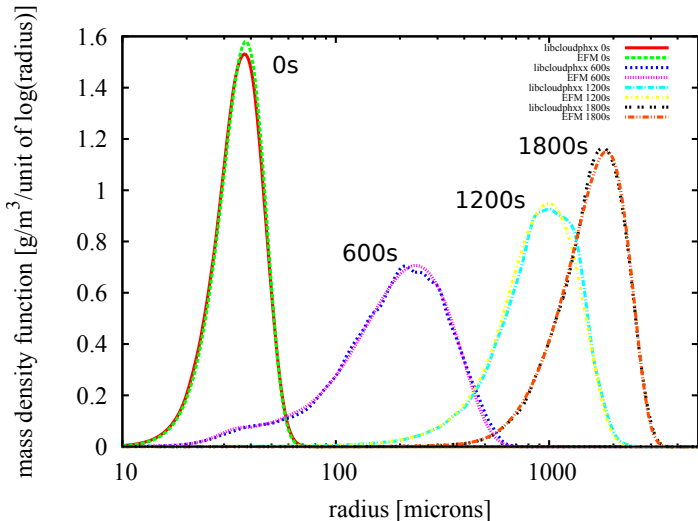


volume:  $10^6 \text{ m}^3$   
LWC:  $1 \text{ g}/\text{m}^3$   
super-droplets:  $2 \times 10^5$

# Coalescence module - current status

- ▶ implemented kernels
  - ▶ Golovin
  - ▶ Geometric
  - ▶ Hall
  - ▶ Hall and Davis
- ▶ terminal velocity formulas
  - ▶ Beard
  - ▶ Khvorostyanov
- ▶ execution policies
  - ▶ CPU (serial or OpenMP for multi-core CPUs)
  - ▶ GPU (CUDA for nVidia hardware)

# Coalescence validation - Hall kernel (CPU)

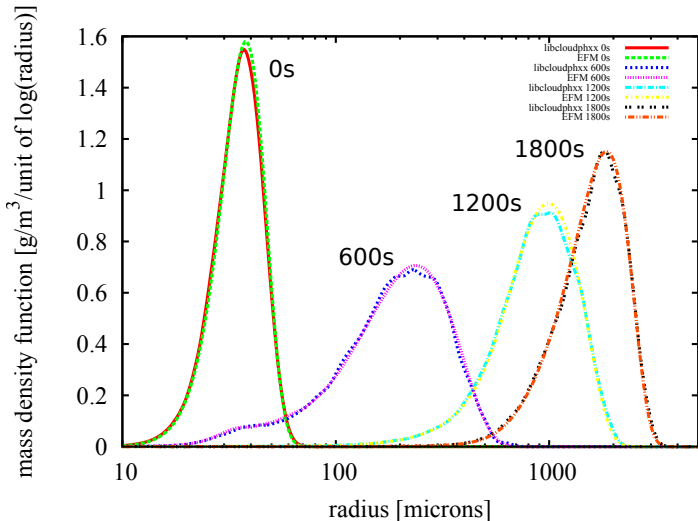


volume:  $10^6 \text{ m}^3$   
LWC:  $1 \text{ g/m}^3$   
super-droplets:  $2 \times 10^5$

# Coalescence module - current status

- ▶ implemented kernels
  - ▶ Golovin
  - ▶ Geometric
  - ▶ Hall
  - ▶ Hall and Davis
- ▶ terminal velocity formulas
  - ▶ Beard
  - ▶ Khvorostyanov
- ▶ execution policies
  - ▶ CPU (serial or OpenMP for multi-core CPUs)
  - ▶ GPU (CUDA for nVidia hardware)

# Coalescence validation - Hall kernel (GPU)



volume:  $10^6 \text{ m}^3$   
LWC:  $1 \text{ g/m}^3$   
super-droplets:  $2 \times 10^5$

# Plan of the talk

- design choices and their rationale
- example results with 2D kinematic set-up
- validation of libcloudph++'s Super-Droplet scheme
- summary & future plans

# Plan of the talk

- design choices and their rationale
- example results with 2D kinematic set-up
- validation of libcloudph++'s Super-Droplet scheme
- summary & future plans



## libcloudph++: summary

### key features:

- ▶ three schemes (all written from scratch):
  - ▶ 1-moment: Kessler
  - ▶ 2-moment: Morrison & Grabowski 2008
  - ▶ Lagrangian: Shima et al. 2009 (Monte-Carlo coalescence)

## libcloudph++: summary

### key features:

- ▶ three schemes (all written from scratch):
  - ▶ 1-moment: Kessler
  - ▶ 2-moment: Morrison & Grabowski 2008
  - ▶ Lagrangian: Shima et al. 2009 (Monte-Carlo coalescence)
- ▶ Lagrangian scheme optionally GPU-resident (via Thrust)

## libcloudph++: summary

### key features:

- ▶ three schemes (all written from scratch):
  - ▶ 1-moment: Kessler
  - ▶ 2-moment: Morrison & Grabowski 2008
  - ▶ Lagrangian: Shima et al. 2009 (Monte-Carlo coalescence)
- ▶ Lagrangian scheme optionally GPU-resident (via Thrust)
- ▶ compact code (500 / 1000 / 4500 lines of code)

## libcloudph++: summary

### key features:

- ▶ three schemes (all written from scratch):
  - ▶ 1-moment: Kessler
  - ▶ 2-moment: Morrison & Grabowski 2008
  - ▶ Lagrangian: Shima et al. 2009 (Monte-Carlo coalescence)
- ▶ Lagrangian scheme optionally GPU-resident (via Thrust)
- ▶ compact code (500 / 1000 / 4500 lines of code)
- ▶ written using Boost.units – compile-time dimensional analysis

# libcloudph++: summary

## key features:

- ▶ three schemes (all written from scratch):
  - ▶ 1-moment: Kessler
  - ▶ 2-moment: Morrison & Grabowski 2008
  - ▶ Lagrangian: Shima et al. 2009 (Monte-Carlo coalescence)
- ▶ Lagrangian scheme optionally GPU-resident (via Thrust)
- ▶ compact code (500 / 1000 / 4500 lines of code)
- ▶ written using Boost.units – compile-time dimensional analysis
- ▶ reusable:
  - ▶ design: no assumptions on dimensionality or dyn-core type
  - ▶ documentation: API described in the paper/manual
  - ▶ legal/practical matters: open source, GPL, hosted on github

## near-future plans

### maintenance (Sylwester)

- ▶ unit test automation
- ▶ 1.0.0 release (May 2015?)

### coalescence (Piotr)

- ▶ implement additional kernels (Pinsky, Vohl,...)
- ▶ include turbulence effects  
(AW kernel, may require modification of the method)
- ▶ include effect of van der Waals forces (Rogers, 1990)

### aqueous chemistry (Ania)

- ▶ finalise implementation (proof-of-concept simulations done)
- ▶ chemistry-coalescence aerosol-processing study

Thank you for your attention!

- ▶ <http://arxiv.org/abs/1310.1905>
- ▶ <http://geosci-model-dev-discuss.net/7/8275/>