Introducing libmpdata++ and libcloudph++: reusable software for atmospheric modelling

Sylwester Arabas

cloud-aerosol modelling team @ University of Warsaw
`foss.igf.fuw.edu.pl`

NCAR, Sep. 4th 2014

Introducing libmpdata++ and libcloudph++: reusable software for atmospheric modelling

Sylwester Arabas

cloud-aerosol modelling team @ University of Warsaw
foss.igf.fuw.edu.pl

NCAR, Sep. 4th 2014

cloud-aerosol modelling team @ University of Warsaw

the team (http://foss.igf.fuw.edu.pl/)

- prof. Hanna Pawłowska (group leader)

cloud-aerosol modelling team @ University of Warsaw

the team (http://foss.igf.fuw.edu.pl/)

- prof. Hanna Pawłowska (group leader)
- in Warsaw:
  - Sylwester Arabas (postdoc)
  - Anna Jaruga (PhD student)
  - Maciej Waruszewski (PhD student)
  - Anna Zimniak (MSc student)

cloud-aerosol modelling team @ University of Warsaw

the team (http://foss.igf.fuw.edu.pl/)

- prof. Hanna Pawłowska (group leader)
- in Warsaw:
  - Sylwester Arabas (postdoc)
  - Anna Jaruga (PhD student)
  - Maciej Waruszewski (PhD student)
  - Anna Zimniak (MSc student)
- overseas:
  - prof. Piotr Smolarkiewicz @ ECMWF
  - prof. Wojciech Grabowski @ NCAR
  - Dorota Jarecka @ NCAR (postdoc)

cloud-aerosol modelling team @ University of Warsaw

the team (http://foss.igf.fuw.edu.pl/)

- prof. Hanna Pawłowska (group leader)
- in Warsaw:
  - Sylwester Arabas (postdoc)
  - Anna Jaruga (PhD student)
  - Maciej Waruszewski (PhD student)
  - Anna Zimniak (MSc student)
- overseas:
  - prof. Piotr Smolarkiewicz @ ECMWF
  - prof. Wojciech Grabowski @ NCAR
  - Dorota Jarecka @ NCAR (postdoc)

aim: developing tools for studying aerosol-cloud interactions

aerosol-cloud interactions: a conceptual picture

aerosol-cloud interactions: a conceptual picture

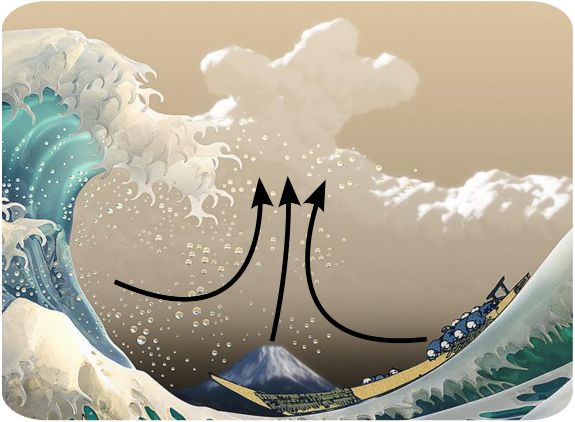
background image: vitsly.ru / Hokusai

aerosol-cloud interactions: a conceptual picture

background image: `vitsly.ru` / Hokusai

aerosol-cloud interactions: a conceptual picture

- aerosol particles of natural and anthropogenic origin act as condensation nuclei



background image: vitsly.ru / Hokusai

aerosol-cloud interactions: a conceptual picture

- aerosol particles of natural and anthropogenic origin act as condensation nuclei
- cloud droplets grow by water vapour condensation

background image: vitsly.ru / Hokusai

aerosol-cloud interactions: a conceptual picture

- aerosol particles of natural and anthropogenic origin act as condensation nuclei
- cloud droplets grow by water vapour condensation
- rain drops form through collisions of cloud droplets

background image: vitsly.ru / Hokusai

aerosol-cloud interactions: a conceptual picture

- aerosol particles of natural and anthropogenic origin act as condensation nuclei
- cloud droplets grow by water vapour condensation
- rain drops form through collisions of cloud droplets
- aqueous chemical reactions irreversibly modify the drop composition

background image: `vitsly.ru` / Hokusai

aerosol-cloud interactions: a conceptual picture

- aerosol particles of natural and anthropogenic origin act as condensation nuclei
- cloud droplets grow by water vapour condensation
- rain drops form through collisions of cloud droplets
- aqueous chemical reactions irreversibly modify the drop composition
- rain drops precipitate washing out aerosol

background image: `vitsly.ru` / Hokusai

aerosol-cloud interactions: a conceptual picture

- aerosol particles of natural and anthropogenic origin act as condensation nuclei
- cloud droplets grow by water vapour condensation
- rain drops form through collisions of cloud droplets
- aqueous chemical reactions irreversibly modify the drop composition
- rain drops precipitate washing out aerosol
- rain drops evaporate into aerosol particles of potentially altered size and/or composition (collisions, chemistry)

background image: vitsly.ru / Hokusai

cloud-aerosol modelling team @ University of Warsaw

the team (http://foss.igf.fuw.edu.pl/)

- prof. Hanna Pawłowska (group leader)
- in Warsaw:
  - Sylwester Arabas (postdoc)
  - Anna Jaruga (PhD student)
  - Maciej Waruszewski (PhD student)
  - Anna Zimniak (MSc student)
- overseas:
  - prof. Piotr Smolarkiewicz @ ECMWF
  - prof. Wojciech Grabowski @ NCAR
  - Dorota Jarecka @ NCAR (postdoc)

aim: developing tools for studying aerosol-cloud interactions

- novel cloud/aerosol microphysics models,

cloud-aerosol modelling team @ University of Warsaw

the team (http://foss.igf.fuw.edu.pl/)

- prof. Hanna Pawłowska (group leader)
- in Warsaw:
  - Sylwester Arabas (postdoc)
  - Anna Jaruga (PhD student)
  - Maciej Waruszewski (PhD student)
  - Anna Zimniak (MSc student)
- overseas:
  - prof. Piotr Smolarkiewicz @ ECMWF
  - prof. Wojciech Grabowski @ NCAR
  - Dorota Jarecka @ NCAR (postdoc)

aim: developing tools for studying aerosol-cloud interactions

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

cloud-aerosol modelling team @ University of Warsaw

the team (http://foss.igf.fuw.edu.pl/)

- prof. Hanna Pawłowska (group leader)
- in Warsaw:
  - Sylwester Arabas (postdoc)
  - Anna Jaruga (PhD student)
  - Maciej Waruszewski (PhD student)
  - Anna Zimniak (MSc student)
- overseas:
  - prof. Piotr Smolarkiewicz @ ECMWF
  - prof. Wojciech Grabowski @ NCAR
  - Dorota Jarecka @ NCAR (postdoc)

aim: developing tools for studying aerosol-cloud interactions

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

free & open-source C++ libraries developed at our group

libmpdata++ / arXiv:1407.1309 / accepted for GMDD

libmpdata++ 0.1: a library of parallel MPDATA solvers
for systems of generalised transport equations

Anna Jaruga[1], Sylwester Arabas[1], Dorota Jarecka[1,2], Hanna Pawlowska[1], Piotr K. Smolarkiewicz[*3], and Maciej Waruszewski[1]

[1]Institute of Geophysics, Faculty of Physics, University of Warsaw, Warsaw, Poland
[2]National Center for Atmospheric Research, Boulder, Colorado, USA
[3]European Centre for Medium-Range Weather Forecasts, Reading, United Kingdom

libcloudph++ / arXiv:1310.1905 / submitted to GMDD

libcloudph++ 0.2: single-moment bulk, double-moment bulk, and particle-based warm-rain microphysics library in C++

Sylwester Arabas[1], Anna Jaruga[1], Hanna Pawlowska[1], Wojciech W. Grabowski[*2]

[1]Institute of Geophysics, Faculty of Physics, University of Warsaw, Warsaw, Poland
[2]National Center for Atmospheric Research (NCAR), Boulder, Colorado, USA

free & open-source C++ libraries developed at our group

libmpdata++ / arXiv:1407.1309 / accepted for GMDD

libmpdata++ 0.1: a library of parallel MPDATA solvers
for systems of generalised transport equations

Anna Jaruga[1], Sylwester Arabas[1], Dorota Jarecka[1,2], Hanna Pawlowska[1], Piotr K. Smolarkiewicz[*3], and Maciej Waruszewski[1]

[1]Institute of Geophysics, Faculty of Physics, University of Warsaw, Warsaw, Poland
[2]National Center for Atmospheric Research, Boulder, Colorado, USA
[3]European Centre for Medium-Range Weather Forecasts, Reading, United Kingdom

libcloudph++ / arXiv:1310.1905 / submitted to GMDD

libcloudph++ 0.2: single-moment bulk, double-moment bulk, and particle-based warm-rain microphysics library in C++

Sylwester Arabas[1], Anna Jaruga[1], Hanna Pawlowska[1], Wojciech W. Grabowski[*2]

[1]Institute of Geophysics, Faculty of Physics, University of Warsaw, Warsaw, Poland
[2]National Center for Atmospheric Research (NCAR), Boulder, Colorado, USA

libmpdata++: 1D advection / hello world

$$\partial_t(G\psi) + \nabla \cdot (G\vec{u}\psi) = GR$$

libmpdata++: 1D advection / hello world

$$\partial_t(G\psi) + \nabla \cdot (G\vec{u}\psi) = GR$$

Smolarkiewicz's MPDATA scheme

- sign-preserving
- non-oscillatory
- small implicit diffusion

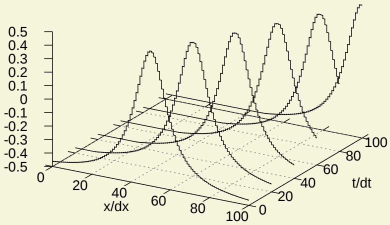
libmpdata++: 1D advection / hello world

$$\partial_t(G\psi) + \nabla \cdot (G\vec{u}\psi) = GR$$

Smolarkiewicz's MPDATA scheme

- sign-preserving
- non-oscillatory
- small implicit diffusion

a "hello-world" example



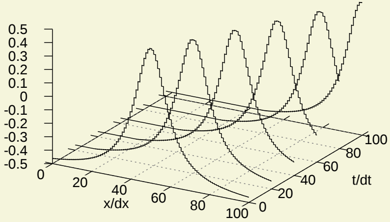
libmpdata++: 1D advection / hello world

$$\partial_t(G\psi) + \nabla \cdot (G\vec{u}\psi) = GR$$

Smolarkiewicz's MPDATA scheme

- sign-preserving
- non-oscillatory
- small implicit diffusion

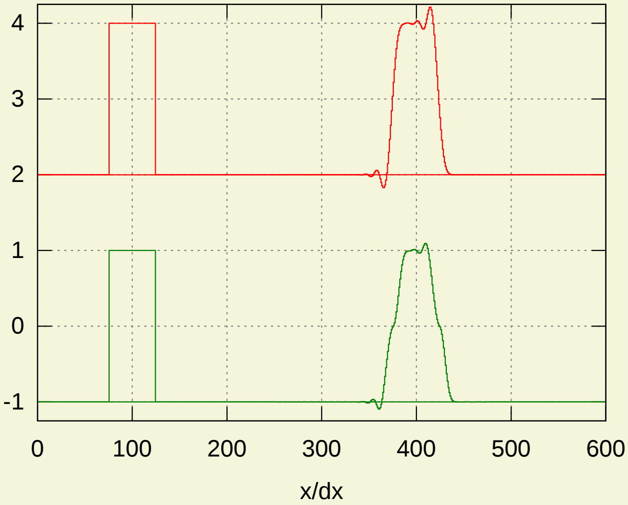
a "hello-world" example




```cpp
#include <libmpdata++/solvers/mpdata.hpp>
#include <libmpdata++/concurr/serial.hpp>
#include <libmpdata++/output/gnuplot.hpp>

using namespace libmpdataxx;
using namespace blitz::tensor;

int main()
{
  // compile-time parameters
  struct ct_params_t : ct_params_default_t
  {
    using real_t = double;
    enum { n_dims = 1 };
    enum { n_eqns = 1 };
  };

  // solver choice
  using slv_t = solvers::mpdata<ct_params_t>;

  // output choice
  using slv_out_t = output::gnuplot<slv_t>;

  // concurency choice + boundary conditions
  using run_t = concurr::serial<slv_out_t,
    bcond::open, bcond::open>;

  // run-time parameters
  typename slv_out_t::rt_params_t p;
  int nx = 101, nt = 100;
  ct_params_t::real_t dx = 0.1;
  p.grid_size = { nx };
  p.outfreq = 20;

  run_t run(p);                  // instantiation
  run.advectee() =               // initial cond.
    -.5 + 1 / (pow(dx*(i - (nx-1)/2.), 2) + 1);
  run.advector() = .5;           // Courant number
  run.advance(nt);               // integration
}
```

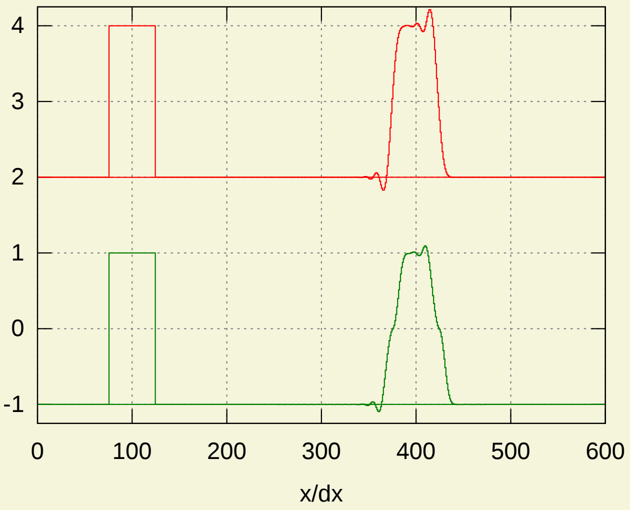
libmpdata++: implemented MPDATA flavours

basic 2-pass MPDATA + "abs" trick for var-sign signals



libmpdata++: implemented MPDATA flavours

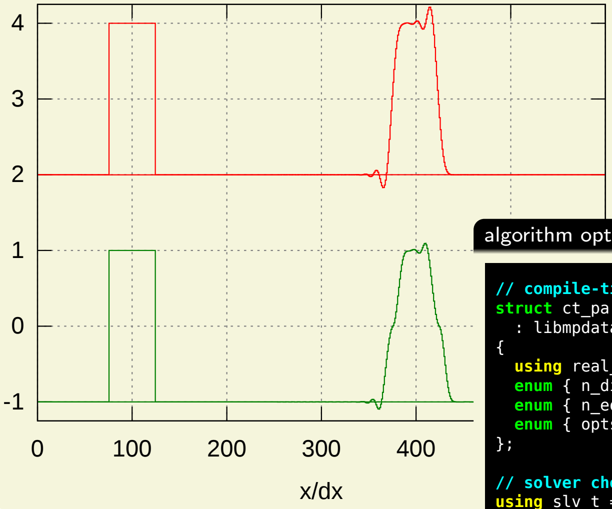
basic 2-pass MPDATA + "abs" trick for var-sign signals



$$\frac{\psi_a-\psi_b}{\psi_a+\psi_b} \approx \frac{|\psi_a|-|\psi_b|}{|\psi_a|+|\psi_b|}$$

libmpdata++: implemented MPDATA flavours

basic 2-pass MPDATA + “abs” trick for var-sign signals

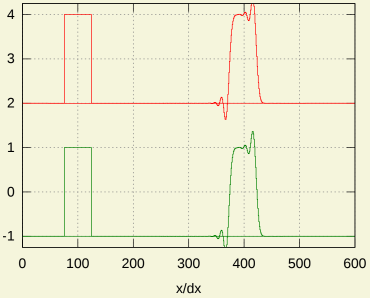


$$\frac{\psi_a - \psi_b}{\psi_a + \psi_b} \approx \frac{|\psi_a| - |\psi_b|}{|\psi_a| + |\psi_b|}$$

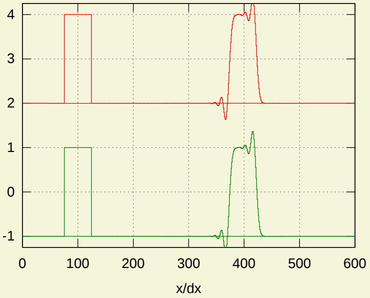
algorithm option as a compile-time parameter

```
// compile-time parameters
struct ct_params_t
  : libmpdataxx::ct_params_default_t
{
  using real_t = double;
  enum { n_dims = 1 };
  enum { n_eqns = 2 };
  enum { opts = opts::abs };
};

// solver choice
using slv_t = solvers::mpdata<ct_params_t>;
```

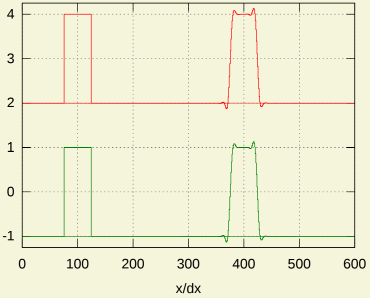
infinite-gauge option ("iga")



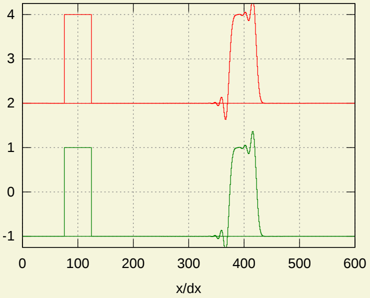
infinite-gauge option ("iga")



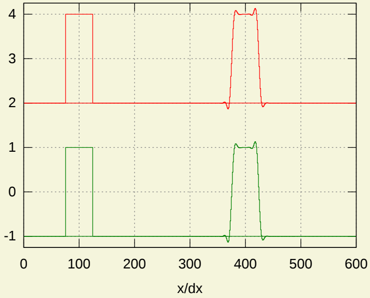
iga + 3rd-order-accurate variant ("tot")



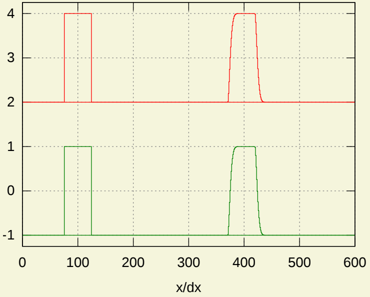
infinite-gauge option ("iga")



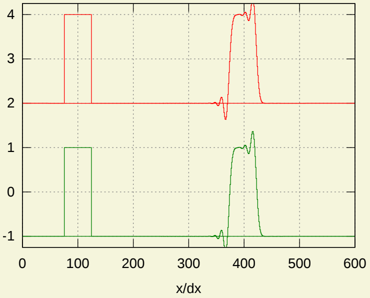
iga + 3rd-order-accurate variant ("tot")



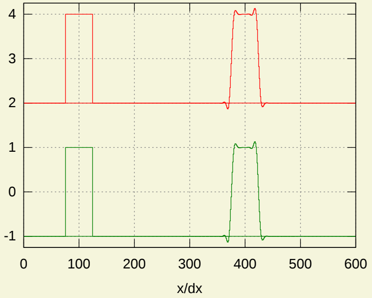
iga + non-oscillatory option ("fct")



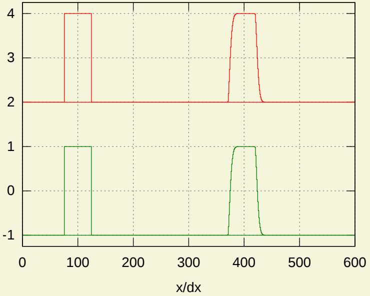
infinite-gauge option ("iga")



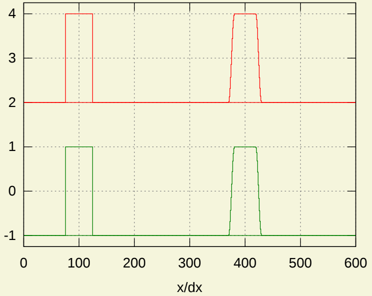
iga + 3rd-order-accurate variant ("tot")



iga + non-oscillatory option ("fct")

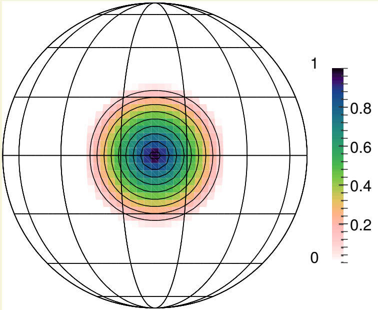


iga + tot + fct

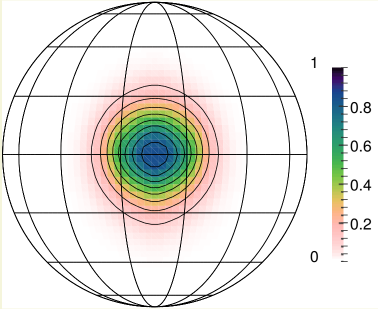


libmpdata++: 2D advection on a sphere

$t = 0$
cone-shaped signal (128×64)
over-the-pole advection

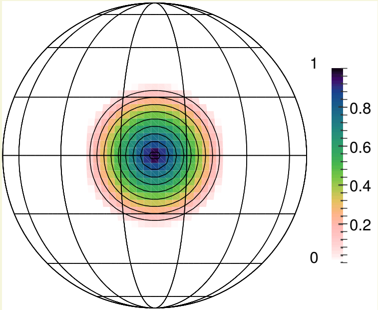


$t/\Delta t = 5120$ (one revolution)
3-pass FCT MPDATA
with third-order terms

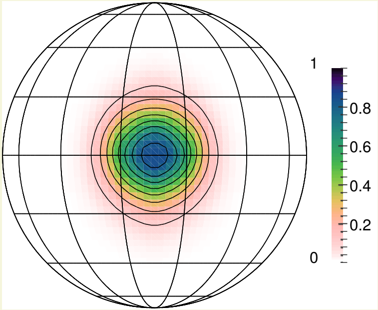


libmpdata++: 2D advection on a sphere

$t = 0$
cone-shaped signal (128×64)
over-the-pole advection



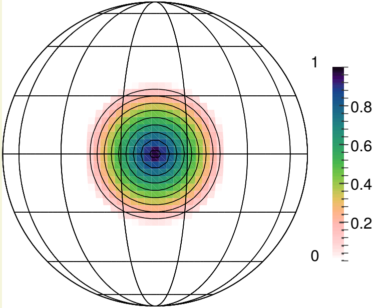
$t/\Delta t = 5120$ (one revolution)
3-pass FCT MPDATA
with third-order terms



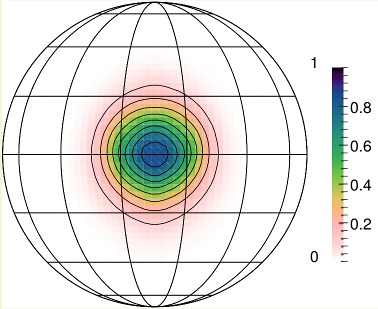
- reproduced experiment of Williamson and Rasch, 1989

libmpdata++: 2D advection on a sphere

$t = 0$
cone-shaped signal (128×64)
over-the-pole advection



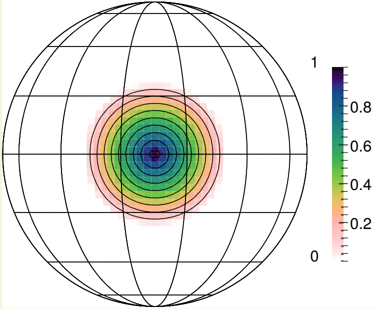
$t/\Delta t = 5120$ (one revolution)
3-pass FCT MPDATA
with third-order terms



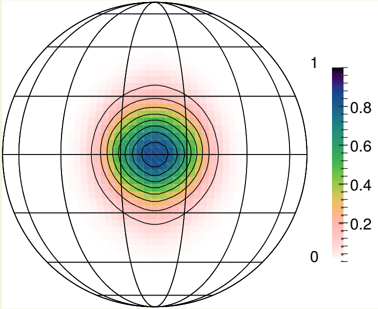
- reproduced experiment of Williamson and Rasch, 1989
- <100 lines of code with libmpdata++

libmpdata++: 2D advection on a sphere

$t = 0$
cone-shaped signal (128×64)
over-the-pole advection



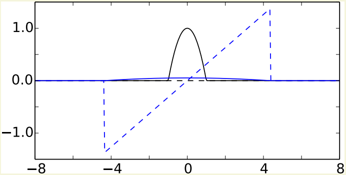
$t/\Delta t = 5120$ (one revolution)
3-pass FCT MPDATA
with third-order terms

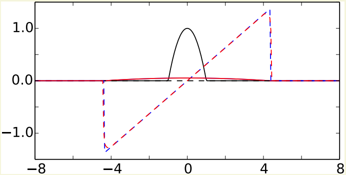


- reproduced experiment of Williamson and Rasch, 1989
- <100 lines of code with libmpdata++
- example developed by Maciej Waruszewski / U. Warsaw

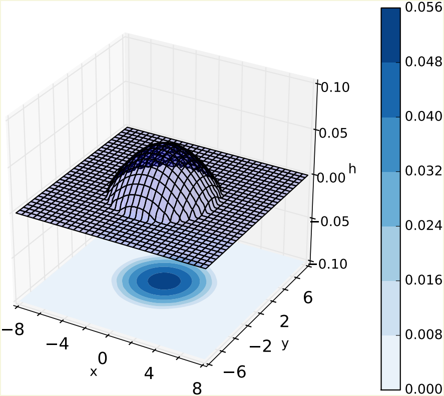
libmpdata++: 2D (3D) shallow-water system

black: initial; blue: analytic; red: MPDATA



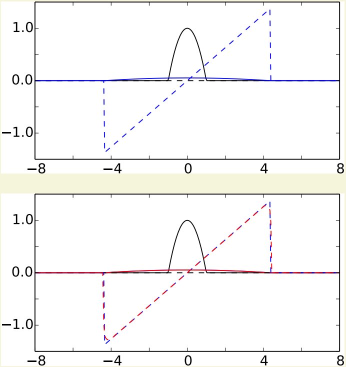


initial condition

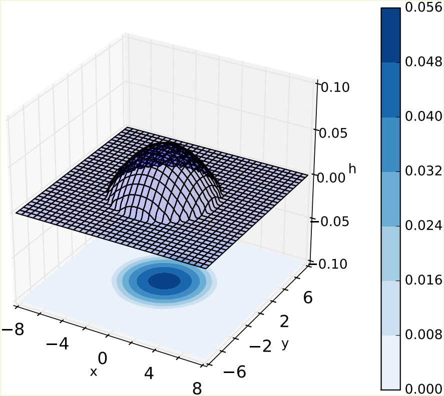


libmpdata++: 2D (3D) shallow-water system

black: initial; blue: analytic;
red: MPDATA



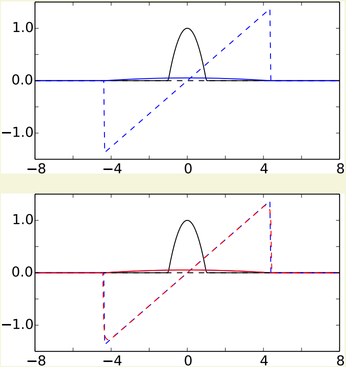
initial condition



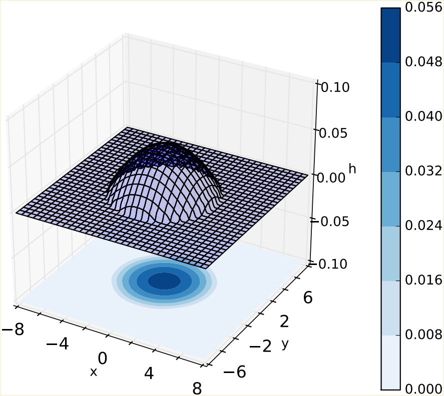
- inspired by 1D experiment of Schär and Smolarkiewicz, 1996

libmpdata++: 2D (3D) shallow-water system

black: initial; blue: analytic;
red: MPDATA



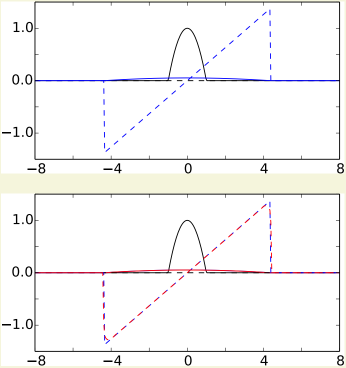
initial condition



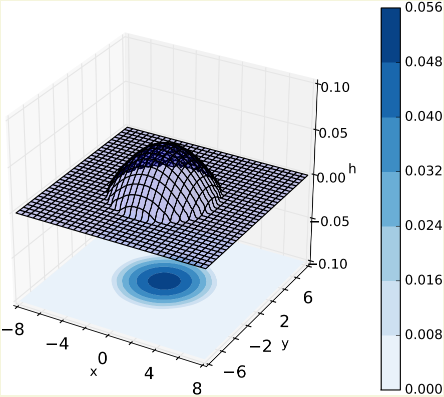
- inspired by 1D experiment of Schär and Smolarkiewicz, 1996
- <300 lines of code with libmpdata++
  (unmodified library, which knows nothing about shallow water eqs)

libmpdata++: 2D (3D) shallow-water system

black: initial; blue: analytic;
red: MPDATA

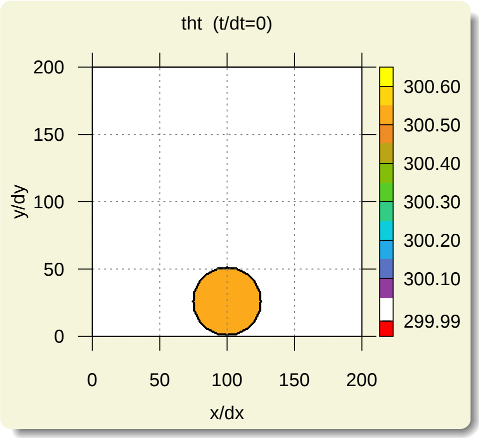


initial condition

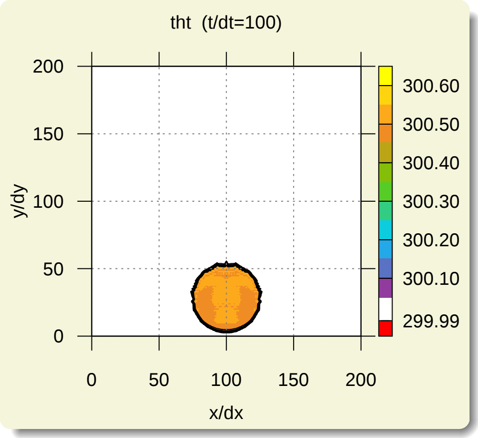


- inspired by 1D experiment of Schär and Smolarkiewicz, 1996
- <300 lines of code with libmpdata++
  (unmodified library, which knows nothing about shallow water eqs)
- example and original analytic solution by Dorota Jarecka / NCAR

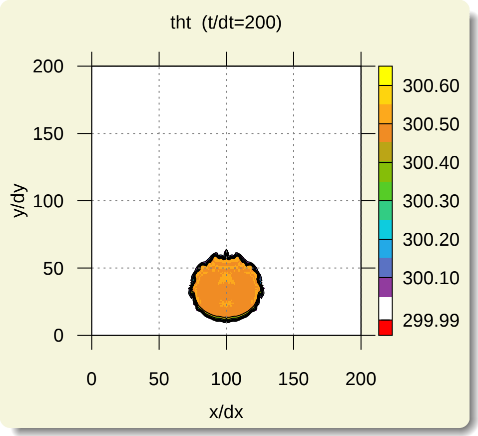
libmpdata++: 2D Boussinesq convection



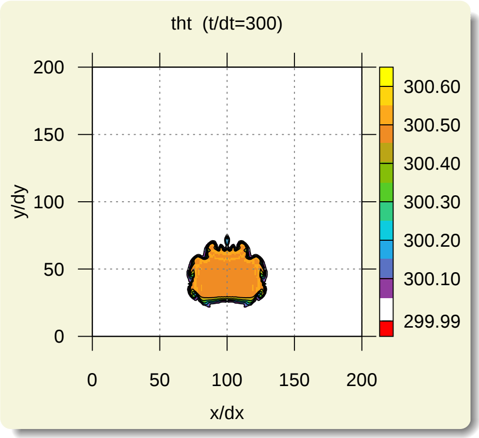
libmpdata++: 2D Boussinesq convection



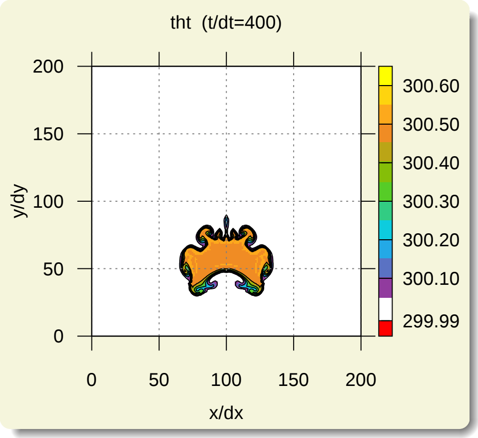
libmpdata++: 2D Boussinesq convection



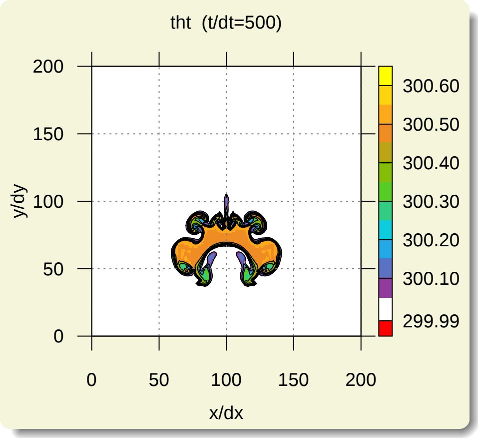
libmpdata++: 2D Boussinesq convection



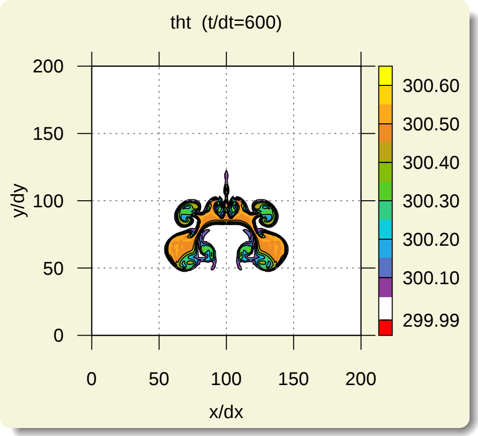
libmpdata++: 2D Boussinesq convection



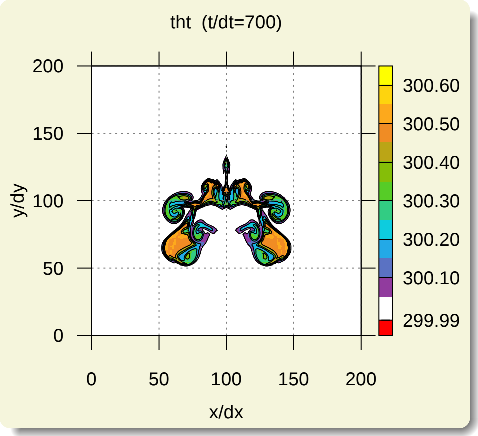
libmpdata++: 2D Boussinesq convection



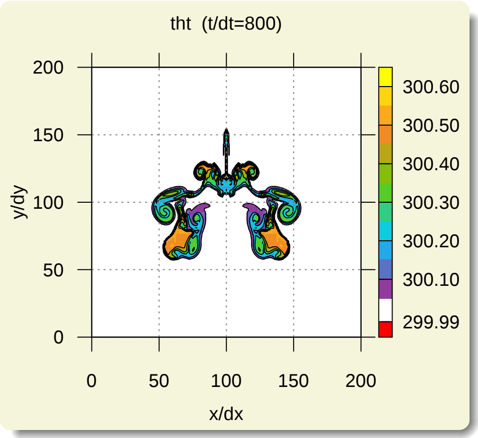
libmpdata++: 2D Boussinesq convection



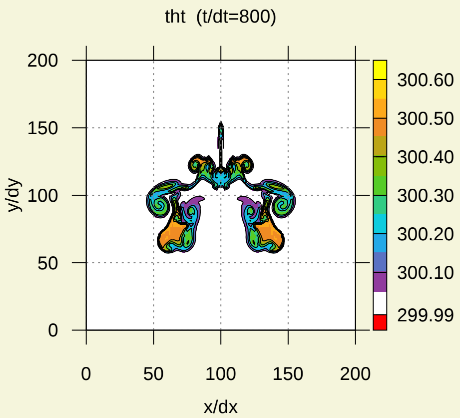
libmpdata++: 2D Boussinesq convection



libmpdata++: 2D Boussinesq convection

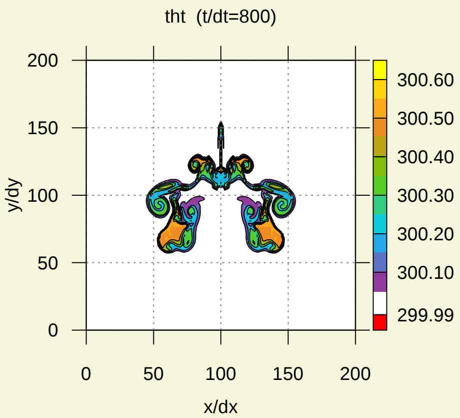


libmpdata++: 2D Boussinesq convection



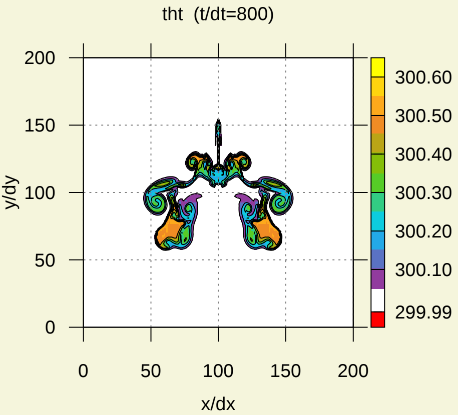
- reproduced experiment of Smolarkiewicz and Pudykiewicz, 1992

libmpdata++: 2D Boussinesq convection



- reproduced experiment of Smolarkiewicz and Pudykiewicz, 1992
- <200 lines of code with libmpdata++
  (using built-in elliptic pressure solver)

libmpdata++: 2D Boussinesq convection



- reproduced experiment of Smolarkiewicz and Pudykiewicz, 1992
- <200 lines of code with libmpdata++
  (using built-in elliptic pressure solver)
- example developed by Anna Jaruga / U. Warsaw

libmpdata++: summary & some technicalities

free & open source C++ library of parallel MPDATA solvers

key features:

- reusable – API documented in the paper; out-of-tree setups
- rich set of MPDATA flavours (incl. FCT, infinite-gauge, ...)
- 1D, 2D & 3D integration; optional coordinate transformation
- four types of solvers:
- implemented using Blitz++ (no loops, expression templates)
- built-in HDF5/XDMF output
- shared-memory parallelisation using OpenMP or Boost.Thread
- separation of concerns (numerics / boundary cond. / io / concurrency)
- compact C++11 code (< 10 kLOC)

libmpdata++: summary & some technicalities

free & open source C++ library of parallel MPDATA solvers

key features:

- reusable – API documented in the paper; out-of-tree setups
- rich set of MPDATA flavours (incl. FCT, infinite-gauge, . . .)
- 1D, 2D & 3D integration; optional coordinate transformation
- four types of solvers:
  - adv (homogeneous advection)
  - adv+rhs (+ right-hand-side terms)
  - adv+rhs+vip (+ prognosed velocity)
  - adv+rhs+vip+prs (+ elliptic pressure solver)
- implemented using Blitz++ (no loops, expression templates)
- built-in HDF5/XDMF output
- shared-memory parallelisation using OpenMP or Boost.Thread
- separation of concerns (numerics / boundary cond. / io / concurrency)
- compact C++11 code (< 10 kLOC)

libmpdata++: summary & some technicalities

free & open source C++ library of parallel MPDATA solvers

key features:

- reusable – API documented in the paper; out-of-tree setups
- rich set of MPDATA flavours (incl. FCT, infinite-gauge, . . .)
- 1D, 2D & 3D integration; optional coordinate transformation
- four types of solvers:
  - adv (homogeneous advection)
  - adv+rhs (+ right-hand-side terms)
  - adv+rhs+vip (+ prognosed velocity)
  - adv+rhs+vip+prs (+ elliptic pressure solver)
- implemented using Blitz++ (no loops, expression templates)
- built-in HDF5/XDMF output
- shared-memory parallelisation using OpenMP or Boost.Thread
- separation of concerns (numerics / boundary cond. / io / concurrency)
- compact C++11 code (< 10 kLOC)

libmpdata++: summary & some technicalities

free & open source C++ library of parallel MPDATA solvers

key features:

- reusable – API documented in the paper; out-of-tree setups
- rich set of MPDATA flavours (incl. FCT, infinite-gauge, . . .)
- 1D, 2D & 3D integration; optional coordinate transformation
- four types of solvers:
  - adv (homogeneous advection)
  - adv+rhs (+ right-hand-side terms)
  - adv+rhs+vip (+ prognosed velocity)
  - adv+rhs+vip+prs (+ elliptic pressure solver)
- implemented using Blitz++ (no loops, expression templates)
- built-in HDF5/XDMF output
- shared-memory parallelisation using OpenMP or Boost.Thread
- separation of concerns (numerics / boundary cond. / io / concurrency)
- compact C++11 code (< 10 kLOC)

libmpdata++: summary & some technicalities

free & open source C++ library of parallel MPDATA solvers

key features:

- reusable – API documented in the paper; out-of-tree setups
- rich set of MPDATA flavours (incl. FCT, infinite-gauge, . . .)
- 1D, 2D & 3D integration; optional coordinate transformation
- four types of solvers:
  - adv (homogeneous advection)
  - adv+rhs (+ right-hand-side terms)
  - adv+rhs+vip (+ prognosed velocity)
  - adv+rhs+vip+prs (+ elliptic pressure solver)
- implemented using Blitz++ (no loops, expression templates)
- built-in HDF5/XDMF output
- shared-memory parallelisation using OpenMP or Boost.Thread
- separation of concerns (numerics / boundary cond. / io / concurrency)
- compact C++11 code (< 10 kLOC)

libmpdata++: summary & some technicalities

free & open source C++ library of parallel MPDATA solvers

key features:

- reusable – API documented in the paper; out-of-tree setups
- rich set of MPDATA flavours (incl. FCT, infinite-gauge, ...)
- 1D, 2D & 3D integration; optional coordinate transformation
- four types of solvers:
  - adv (homogeneous advection)
  - adv+rhs (+ right-hand-side terms)
  - adv+rhs+vip (+ prognosed velocity)
  - adv+rhs+vip+prs (+ elliptic pressure solver)
- implemented using Blitz++ (no loops, expression templates)
- built-in HDF5/XDMF output
- shared-memory parallelisation using OpenMP or Boost.Thread
- separation of concerns (numerics / boundary cond. / io / concurrency)
- compact C++11 code (< 10 kLOC)

libmpdata++: summary & some technicalities

free & open source C++ library of parallel MPDATA solvers

key features:

- reusable – API documented in the paper; out-of-tree setups
- rich set of MPDATA flavours (incl. FCT, infinite-gauge, . . .)
- 1D, 2D & 3D integration; optional coordinate transformation
- four types of solvers:
  - adv (homogeneous advection)
  - adv+rhs (+ right-hand-side terms)
  - adv+rhs+vip (+ prognosed velocity)
  - adv+rhs+vip+prs (+ elliptic pressure solver)
- implemented using Blitz++ (no loops, expression templates)
- built-in HDF5/XDMF output
- shared-memory parallelisation using OpenMP or Boost.Thread
- separation of concerns (numerics / boundary cond. / io / concurrency)
- compact C++11 code (< 10 kLOC)

libmpdata++: summary & some technicalities

free & open source C++ library of parallel MPDATA solvers

key features:

- reusable – API documented in the paper; out-of-tree setups
- rich set of MPDATA flavours (incl. FCT, infinite-gauge, ...)
- 1D, 2D & 3D integration; optional coordinate transformation
- four types of solvers:
  - adv (homogeneous advection)
  - adv+rhs (+ right-hand-side terms)
  - adv+rhs+vip (+ prognosed velocity)
  - adv+rhs+vip+prs (+ elliptic pressure solver)
- implemented using Blitz++ (no loops, expression templates)
- built-in HDF5/XDMF output
- shared-memory parallelisation using OpenMP or Boost.Thread
- separation of concerns (numerics / boundary cond. / io / concurrency)
- compact C++11 code (< 10 kLOC)

libmpdata++: summary & some technicalities

free & open source C++ library of parallel MPDATA solvers

key features:

- reusable – API documented in the paper; out-of-tree setups
- rich set of MPDATA flavours (incl. FCT, infinite-gauge, . . .)
- 1D, 2D & 3D integration; optional coordinate transformation
- four types of solvers:
  - adv (homogeneous advection)
  - adv+rhs (+ right-hand-side terms)
  - adv+rhs+vip (+ prognosed velocity)
  - adv+rhs+vip+prs (+ elliptic pressure solver)
- implemented using Blitz++ (no loops, expression templates)
- built-in HDF5/XDMF output
- shared-memory parallelisation using OpenMP or Boost.Thread
- separation of concerns (numerics / boundary cond. / io / concurrency)
- compact C++11 code (< 10 kLOC)

libmpdata++: summary & some technicalities

free & open source C++ library of parallel MPDATA solvers

key features:

- reusable – API documented in the paper; out-of-tree setups
- rich set of MPDATA flavours (incl. FCT, infinite-gauge, ...)
- 1D, 2D & 3D integration; optional coordinate transformation
- four types of solvers:
  - adv (homogeneous advection)
  - adv+rhs (+ right-hand-side terms)
  - adv+rhs+vip (+ prognosed velocity)
  - adv+rhs+vip+prs (+ elliptic pressure solver)
- implemented using Blitz++ (no loops, expression templates)
- built-in HDF5/XDMF output
- shared-memory parallelisation using OpenMP or Boost.Thread
- separation of concerns (numerics / boundary cond. / io / concurrency)
- compact C++11 code (< 10 kLOC)

libmpdata++: summary & some technicalities

free & open source C++ library of parallel MPDATA solvers

key features:

- reusable – API documented in the paper; out-of-tree setups
- rich set of MPDATA flavours (incl. FCT, infinite-gauge, ...)
- 1D, 2D & 3D integration; optional coordinate transformation
- four types of solvers:
  - adv (homogeneous advection)
  - adv+rhs (+ right-hand-side terms)
  - adv+rhs+vip (+ prognosed velocity)
  - adv+rhs+vip+prs (+ elliptic pressure solver)
- implemented using Blitz++ (no loops, expression templates)
- built-in HDF5/XDMF output
- shared-memory parallelisation using OpenMP or Boost.Thread
- separation of concerns (numerics / boundary cond. / io / concurrency)
- compact C++11 code (< 10 kLOC)

libmpdata++: summary & some technicalities

free & open source C++ library of parallel MPDATA solvers

key features:

- reusable – API documented in the paper; out-of-tree setups
- rich set of MPDATA flavours (incl. FCT, infinite-gauge, ...)
- 1D, 2D & 3D integration; optional coordinate transformation
- four types of solvers:
  - adv (homogeneous advection)
  - adv+rhs (+ right-hand-side terms)
  - adv+rhs+vip (+ prognosed velocity)
  - adv+rhs+vip+prs (+ elliptic pressure solver)
- implemented using Blitz++ (no loops, expression templates)
- built-in HDF5/XDMF output
- shared-memory parallelisation using OpenMP or Boost.Thread
- separation of concerns (numerics / boundary cond. / io / concurrency)
- compact C++11 code (< 10 kLOC)

libmpdata++: summary & some technicalities

free & open source C++ library of parallel MPDATA solvers

key features:

- reusable – API documented in the paper; out-of-tree setups
- rich set of MPDATA flavours (incl. FCT, infinite-gauge, ...)
- 1D, 2D & 3D integration; optional coordinate transformation
- four types of solvers:
  - adv (homogeneous advection)
  - adv+rhs (+ right-hand-side terms)
  - adv+rhs+vip (+ prognosed velocity)
  - adv+rhs+vip+prs (+ elliptic pressure solver)
- implemented using Blitz++ (no loops, expression templates)
- built-in HDF5/XDMF output
- shared-memory parallelisation using OpenMP or Boost.Thread
- separation of concerns (numerics / boundary cond. / io / concurrency)
- compact C++11 code (< 10 kLOC)

libmpdata++: summary & some technicalities

free & open source C++ library of parallel MPDATA solvers

key features:

- reusable – API documented in the paper; out-of-tree setups
- rich set of MPDATA flavours (incl. FCT, infinite-gauge, ...)
- 1D, 2D & 3D integration; optional coordinate transformation
- four types of solvers:
  - adv (homogeneous advection)
  - adv+rhs (+ right-hand-side terms)
  - adv+rhs+vip (+ prognosed velocity)
  - adv+rhs+vip+prs (+ elliptic pressure solver)
- implemented using Blitz++ (no loops, expression templates)
- built-in HDF5/XDMF output
- shared-memory parallelisation using OpenMP or Boost.Thread
- separation of concerns (numerics / boundary cond. / io / concurrency)
- compact C++11 code (< 10 kLOC)

free & open-source C++ libraries developed at our group

libmpdata++ / arXiv:1407.1309 / accepted for GMDD

libmpdata++ 0.1: a library of parallel MPDATA solvers for systems of generalised transport equations

Anna Jaruga[1], Sylwester Arabas[1], Dorota Jarecka[1,2], Hanna Pawlowska[1], Piotr K. Smolarkiewicz[*3], and Maciej Waruszewski[1]

[1]Institute of Geophysics, Faculty of Physics, University of Warsaw, Warsaw, Poland
[2]National Center for Atmospheric Research, Boulder, Colorado, USA
[3]European Centre for Medium-Range Weather Forecasts, Reading, United Kingdom

libcloudph++ / arXiv:1310.1905 / submitted to GMDD

libcloudph++ 0.2: single-moment bulk, double-moment bulk, and particle-based warm-rain microphysics library in C++

Sylwester Arabas[1], Anna Jaruga[1], Hanna Pawlowska[1], Wojciech W. Grabowski[*2]

[1]Institute of Geophysics, Faculty of Physics, University of Warsaw, Warsaw, Poland
[2]National Center for Atmospheric Research (NCAR), Boulder, Colorado, USA

free & open-source C++ libraries developed at our group

libmpdata++ / arXiv:1407.1309 / accepted for GMDD

libmpdata++ 0.1: a library of parallel MPDATA solvers for systems of generalised transport equations

Anna Jaruga[1], Sylwester Arabas[1], Dorota Jarecka[1,2], Hanna Pawlowska[1], Piotr K. Smolarkiewicz*[3], and Maciej Waruszewski[1]

[1]Institute of Geophysics, Faculty of Physics, University of Warsaw, Warsaw, Poland
[2]National Center for Atmospheric Research, Boulder, Colorado, USA
[3]European Centre for Medium-Range Weather Forecasts, Reading, United Kingdom

libcloudph++ / arXiv:1310.1905 / submitted to GMDD

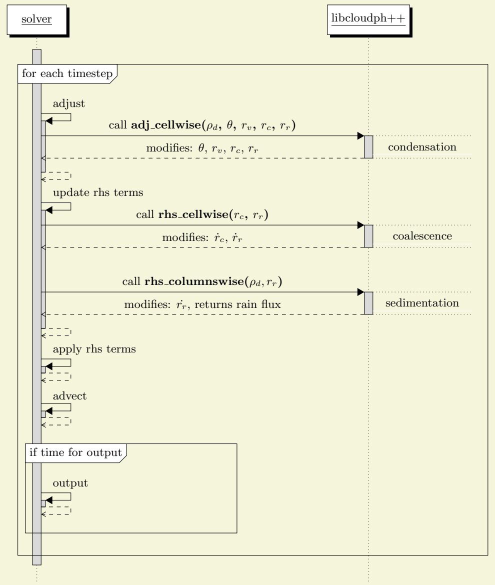
libcloudph++ 0.2: single-moment bulk, double-moment bulk, and particle-based warm-rain microphysics library in C++

Sylwester Arabas[1], Anna Jaruga[1], Hanna Pawlowska[1], Wojciech W. Grabowski*[2]

[1]Institute of Geophysics, Faculty of Physics, University of Warsaw, Warsaw, Poland
[2]National Center for Atmospheric Research (NCAR), Boulder, Colorado, USA

libcloudph++: reusable μ-physics

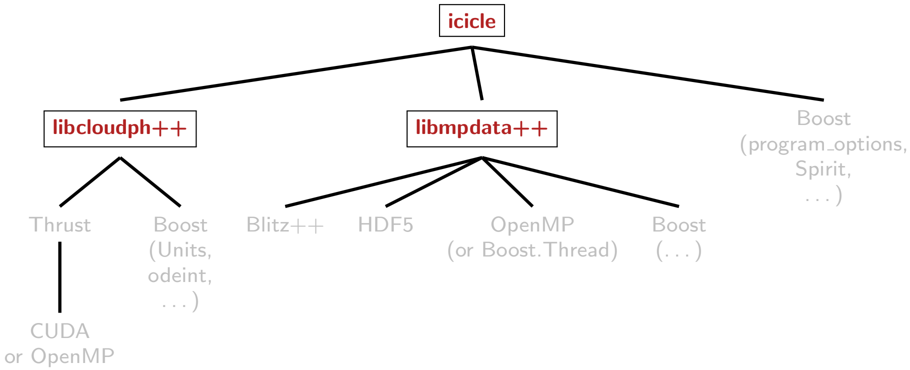
sequence diagram



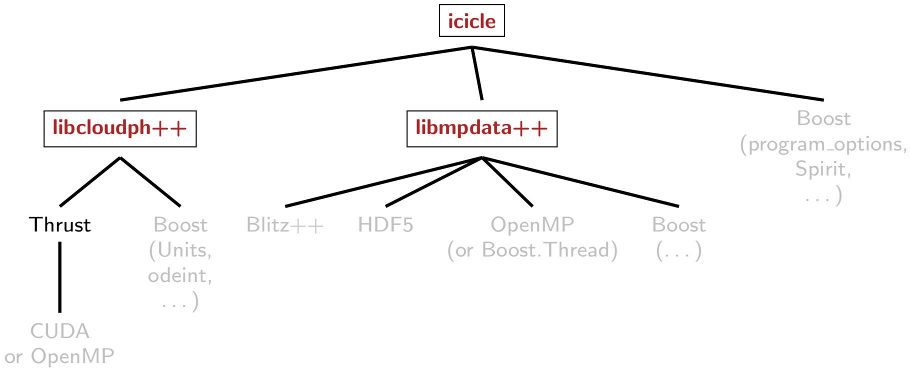
programming interface

```
template<typename real_t>
struct opts_t
{
  bool
    acti = true, // activation
    cond = true, // condensation
    acnv = true, // autoconversion
    accr = true, // accretion
    sedi = true; // sedimentation

  // RH limit for activation
  real_t RH_max = 44;

  // aerosol spectrum
  struct lognormal_mode_t
  {
    real_t
      mean_rd,   // [m]
      sdev_rd,   // [1]
      N_stp,     // [m-3] @STP
      chem_b;    // [1]
  };
  std::vector<lognormal_mode_t> dry_distros;
};
```

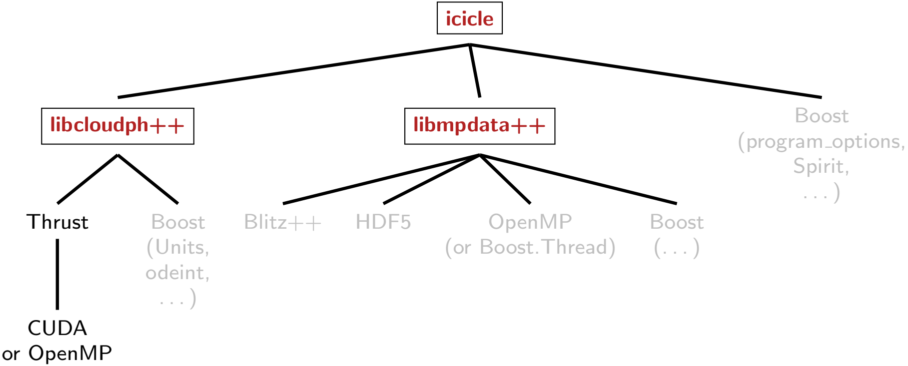
libcloudph++: example coupling with libmpdata++



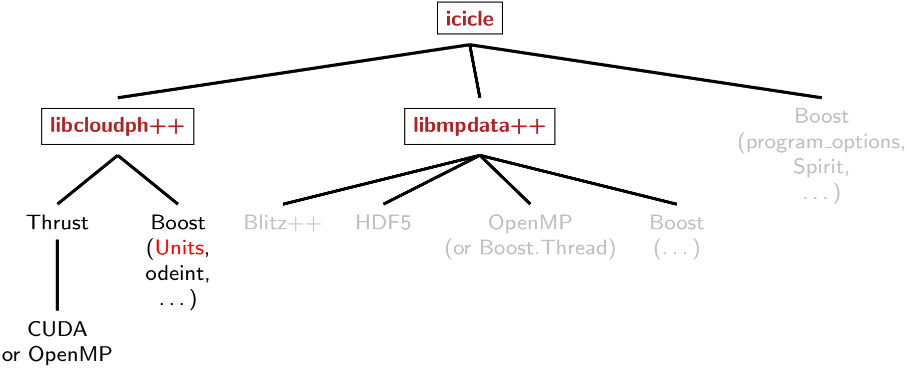
libcloudph++: example coupling with libmpdata++



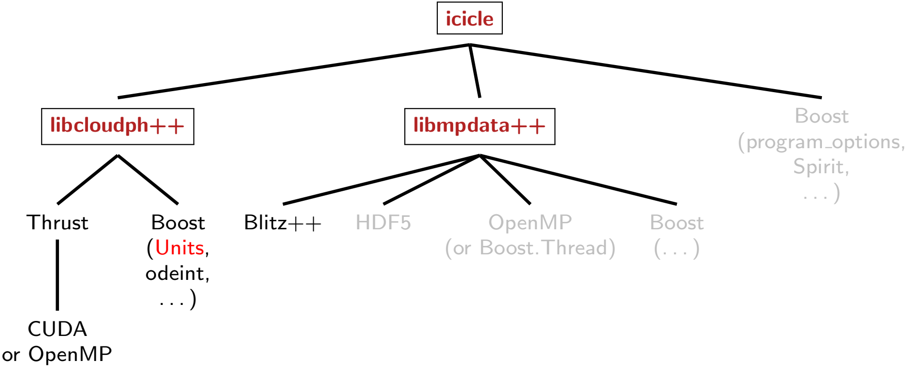
libcloudph++: example coupling with libmpdata++



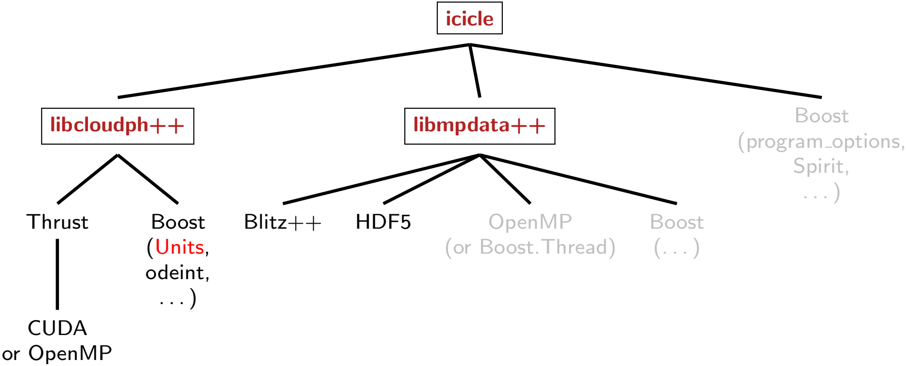
libcloudph++: example coupling with libmpdata++



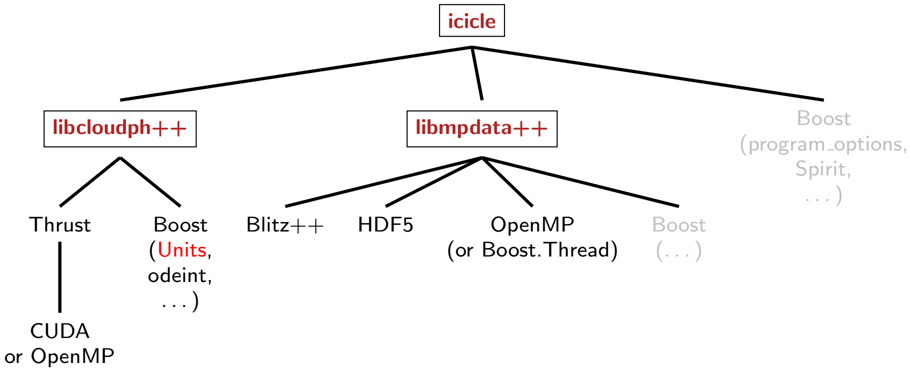
libcloudph++: example coupling with libmpdata++



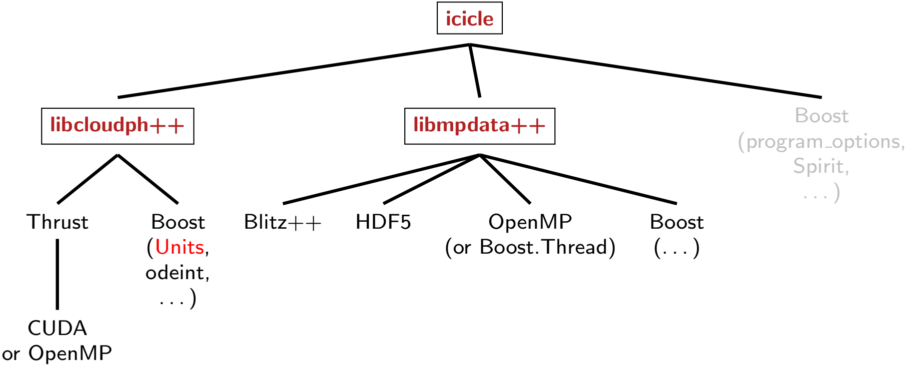
libcloudph++: example coupling with libmpdata++



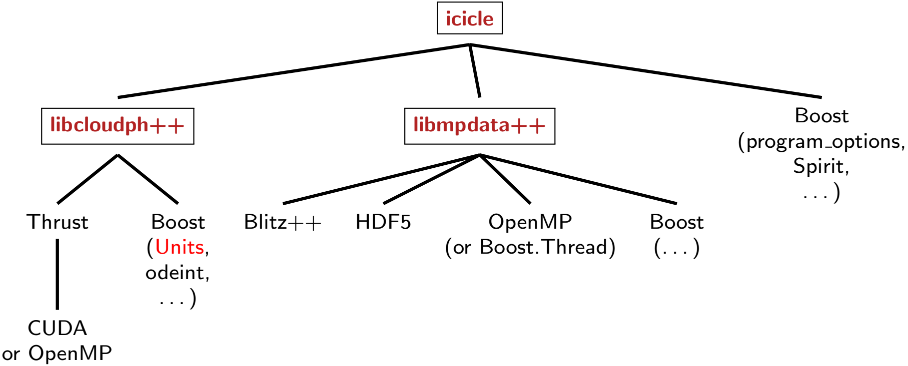
libcloudph++: example coupling with libmpdata++



libcloudph++: example coupling with libmpdata++

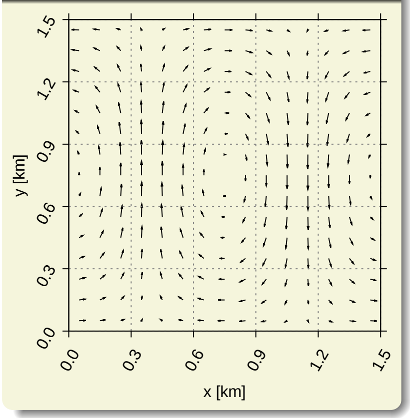


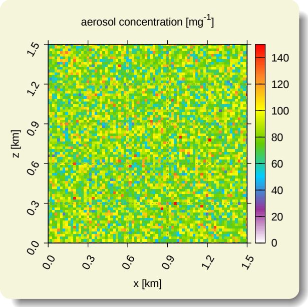
libcloudph++: example coupling with libmpdata++



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

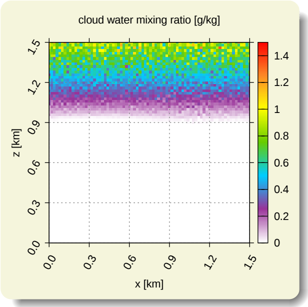
single-eddy velocity field

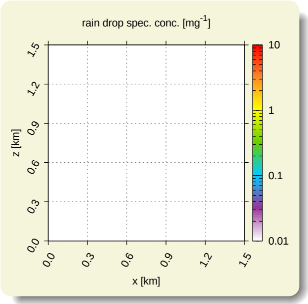


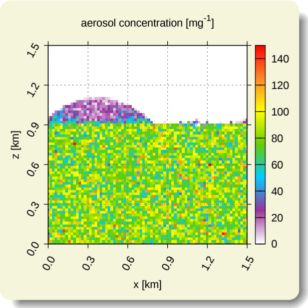


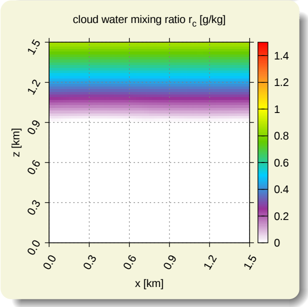
- set-up: Grabowski & Lebo (ICMW 2012)
- 2D prescribed flow
- advection: libmpdata++ (2-pass FCT)
- μ-physics: libcloudph++

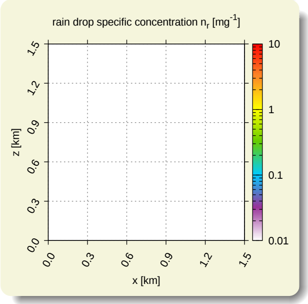
libcloudph++: VOCALS-inspired aerosol processing set-up
x









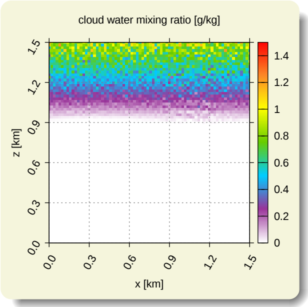


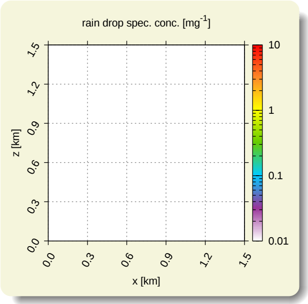
Lagrangian/Monte-Carlo scheme
(Shima et al. 2009)

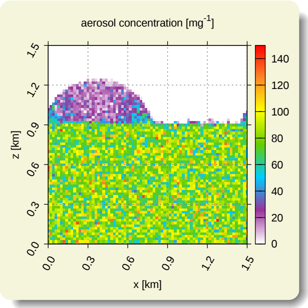
2-moment bulk scheme
(Morrison & Grabowski 2007)

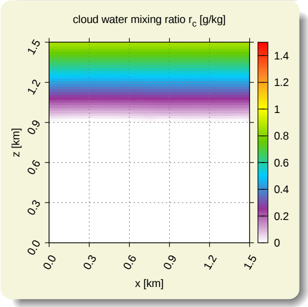
libcloudph++: VOCALS-inspired aerosol processing set-up

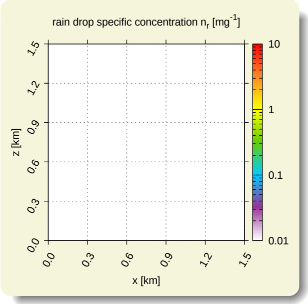
x x









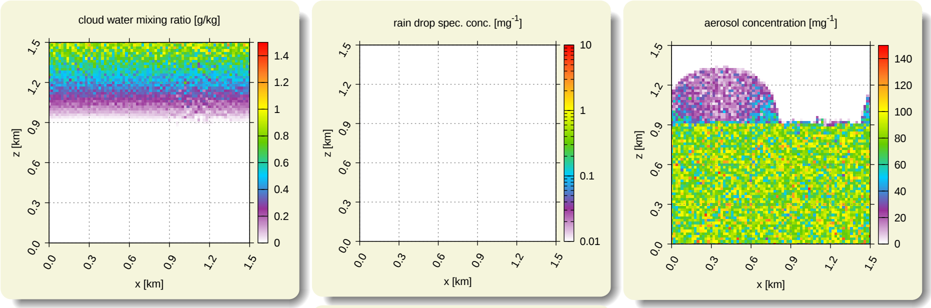


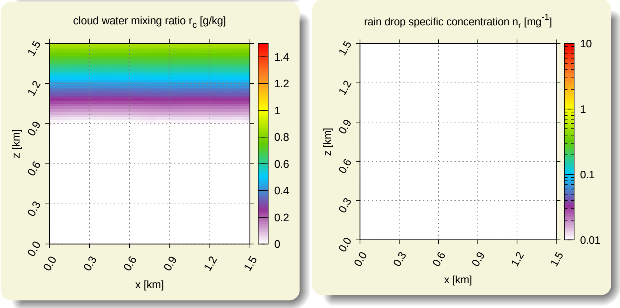
Lagrangian/Monte-Carlo scheme
(Shima et al. 2009)

2-moment bulk scheme
(Morrison & Grabowski 2007)

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

xxx



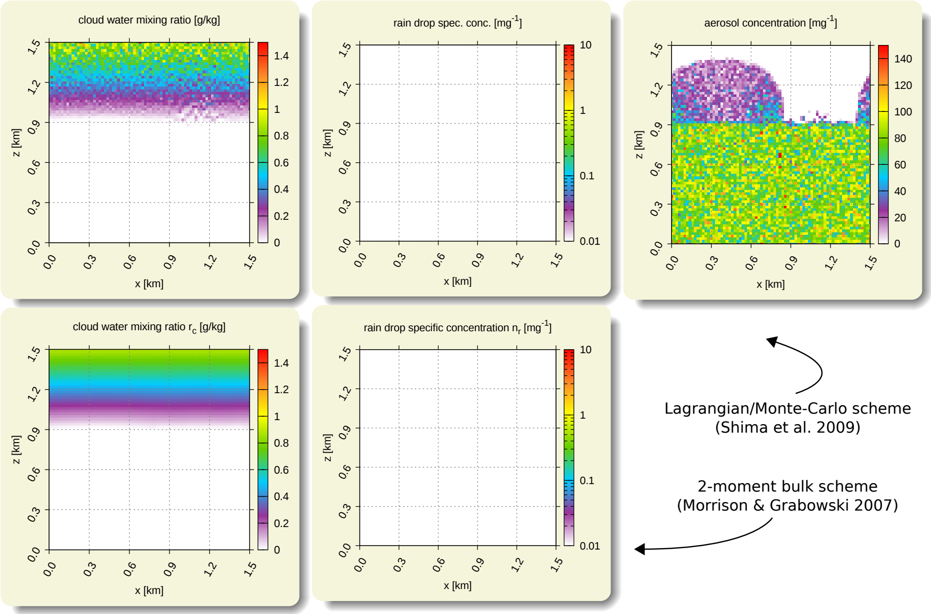


Lagrangian/Monte-Carlo scheme
(Shima et al. 2009)

2-moment bulk scheme
(Morrison & Grabowski 2007)

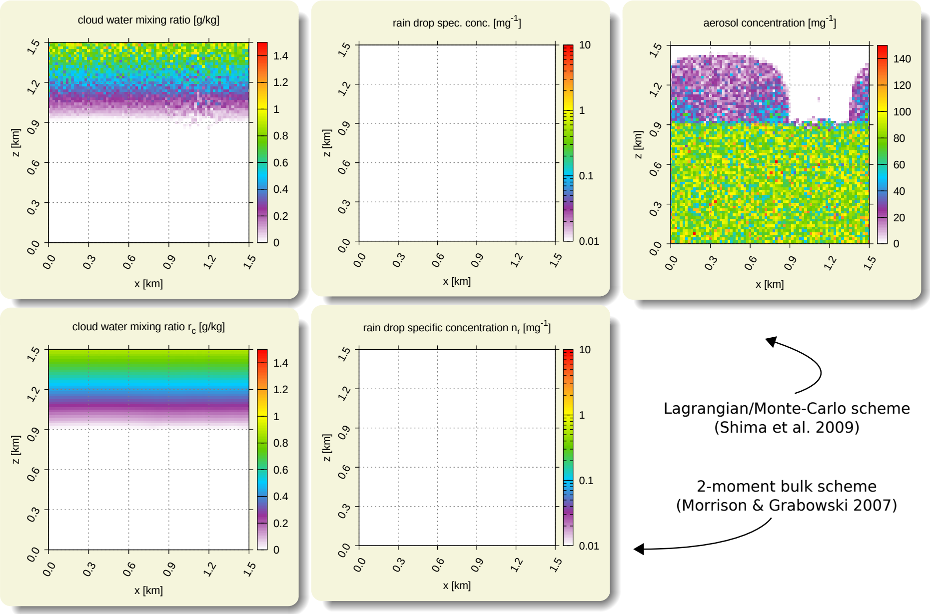
libcloudph++: VOCALS-inspired aerosol processing set-up

xxxx



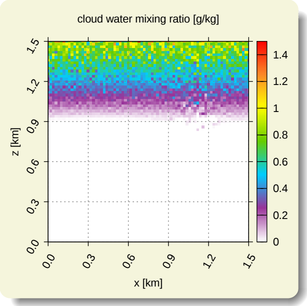
libcloudph++: VOCALS-inspired aerosol processing set-up

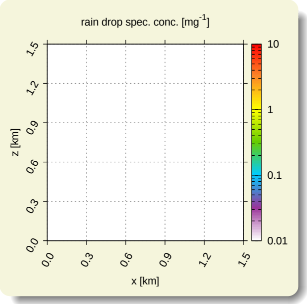
XXXXX

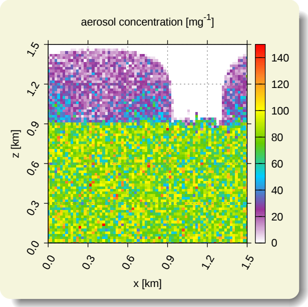


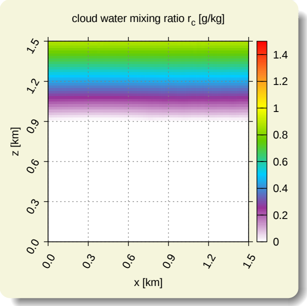
libcloudph++: VOCALS-inspired aerosol processing set-up

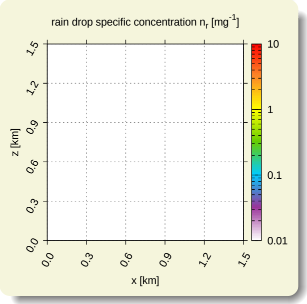
xxxxxx









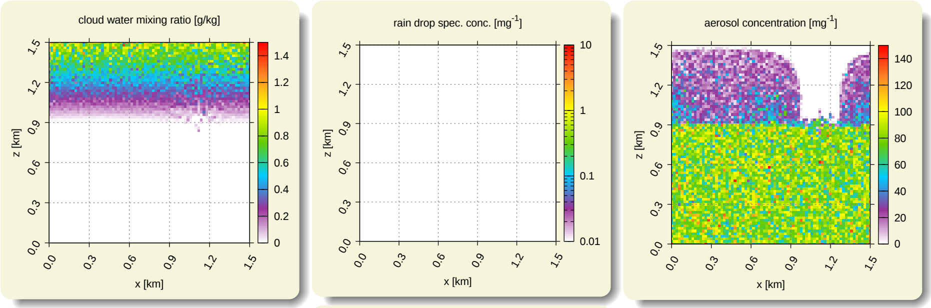


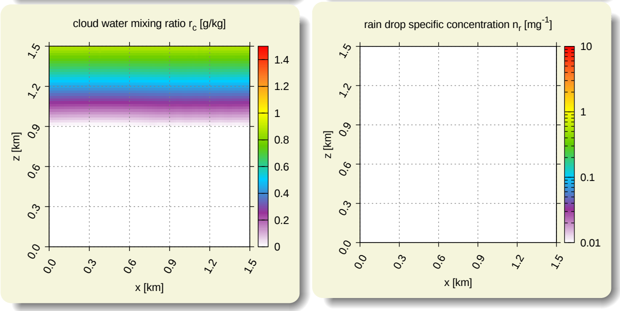
Lagrangian/Monte-Carlo scheme
(Shima et al. 2009)

2-moment bulk scheme
(Morrison & Grabowski 2007)

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

xxxxxxx



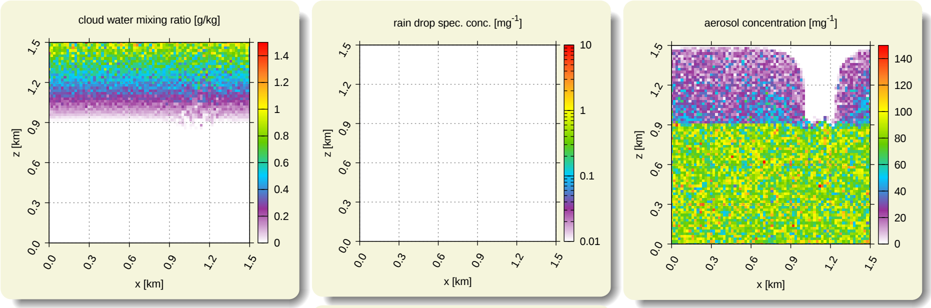


Lagrangian/Monte-Carlo scheme (Shima et al. 2009)

2-moment bulk scheme (Morrison & Grabowski 2007)

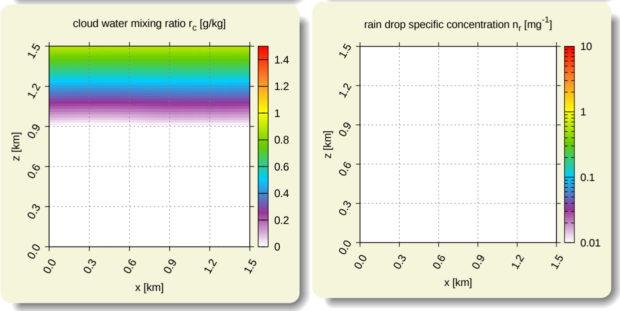
libcloudph++: VOCALS-inspired aerosol processing set-up

xxxxxxxx



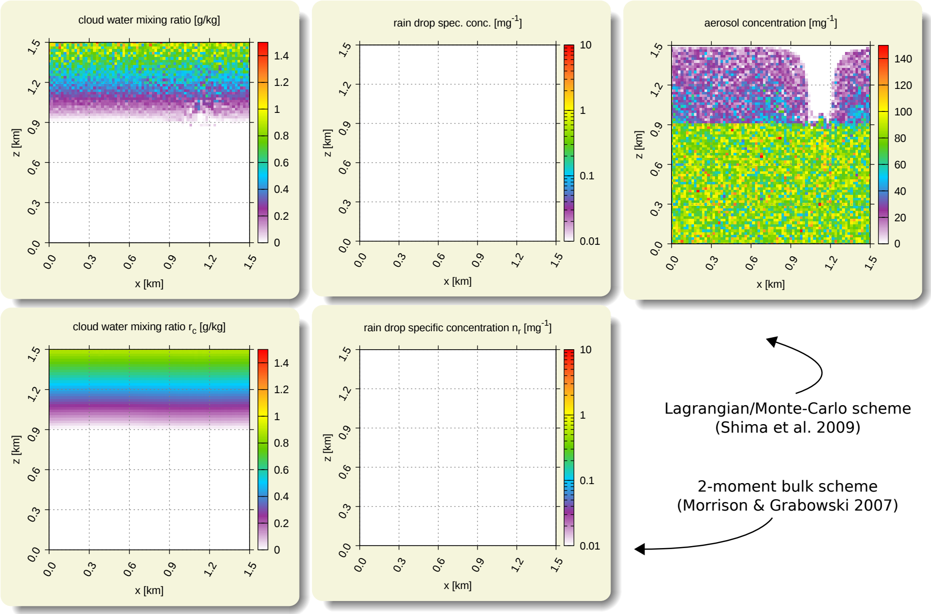
Lagrangian/Monte-Carlo scheme
(Shima et al. 2009)

2-moment bulk scheme
(Morrison & Grabowski 2007)



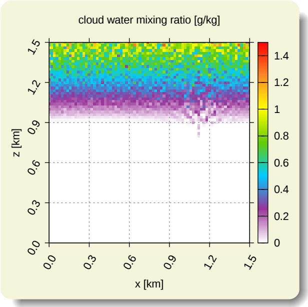
libcloudph++: VOCALS-inspired aerosol processing set-up

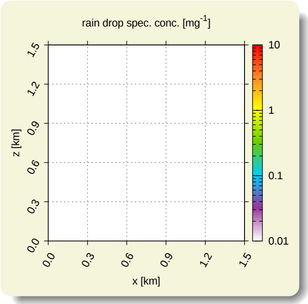
xxxxxxxxx

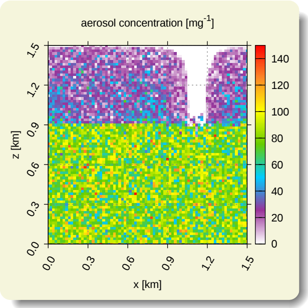


libcloudph++: VOCALS-inspired aerosol processing set-up

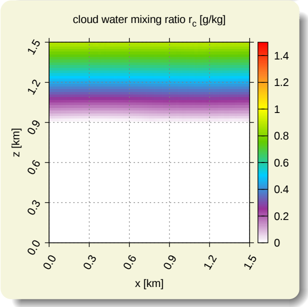
xxxxxxxxxx

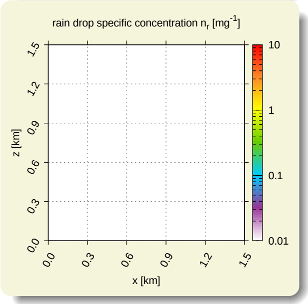






Lagrangian/Monte-Carlo scheme
(Shima et al. 2009)

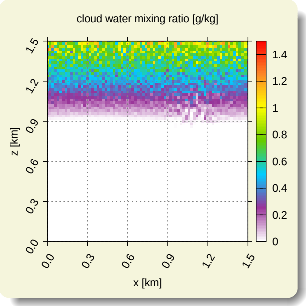


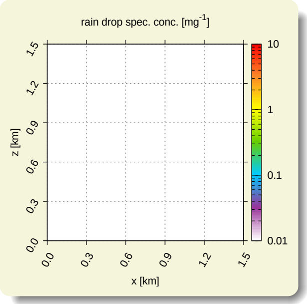


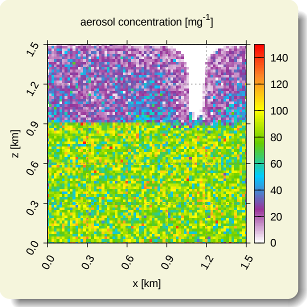
2-moment bulk scheme
(Morrison & Grabowski 2007)

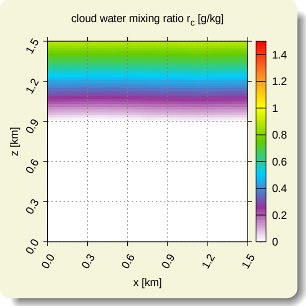
libcloudph++: VOCALS-inspired aerosol processing set-up

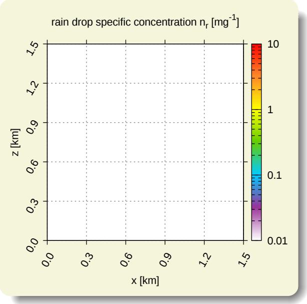
xxxxxxxxxxx









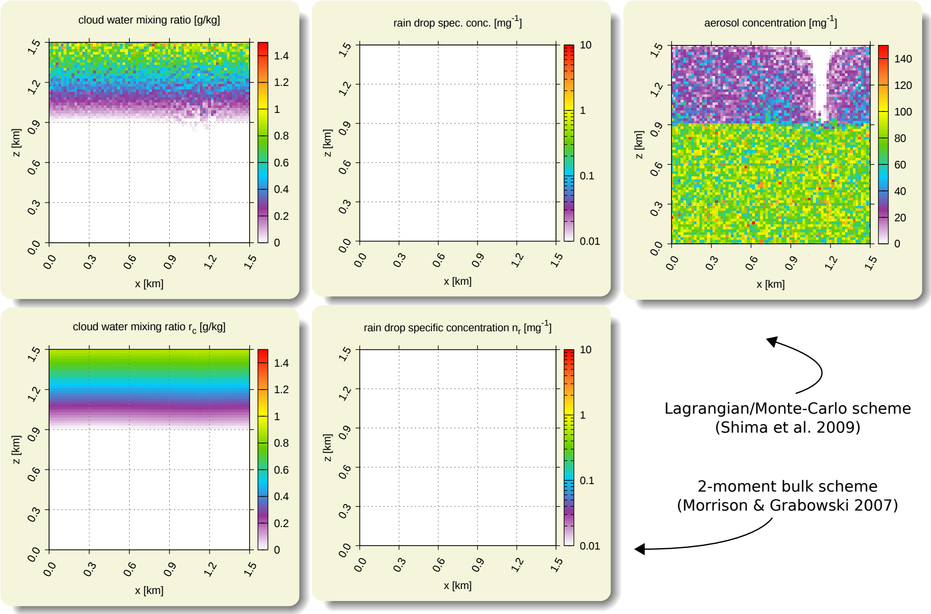


Lagrangian/Monte-Carlo scheme
(Shima et al. 2009)

2-moment bulk scheme
(Morrison & Grabowski 2007)

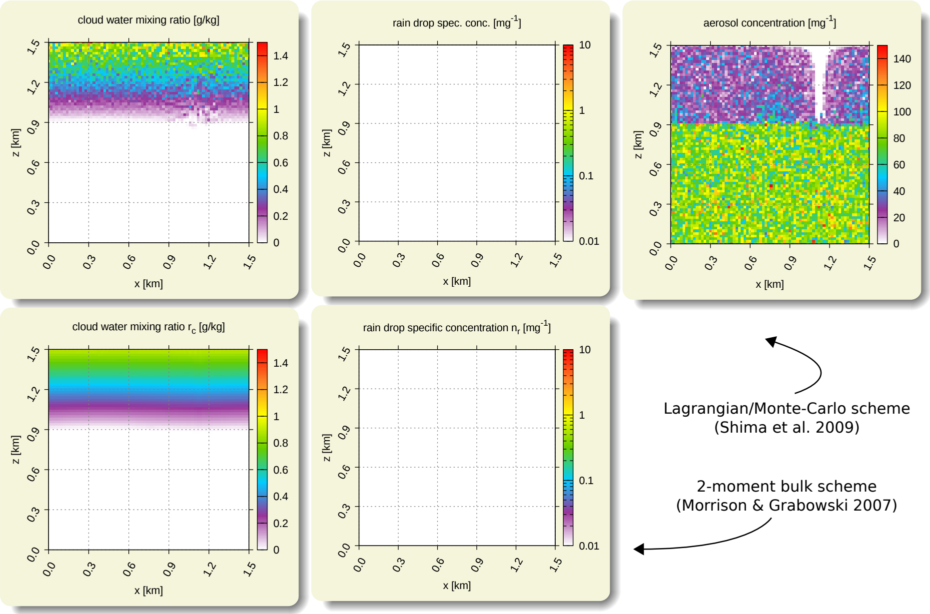
libcloudph++: VOCALS-inspired aerosol processing set-up

XXXXXXXXXXXX



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

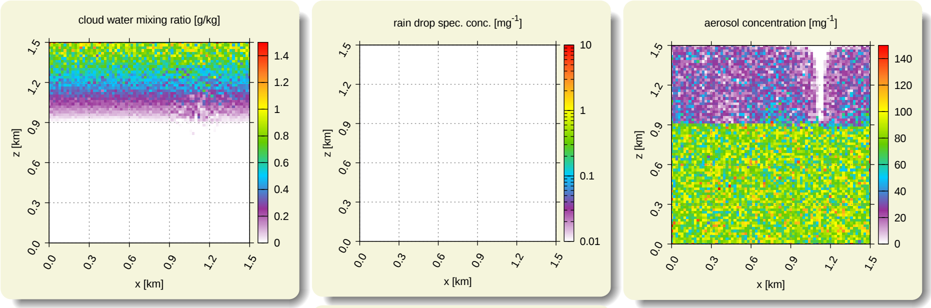
xxxxxxxxxxxxx

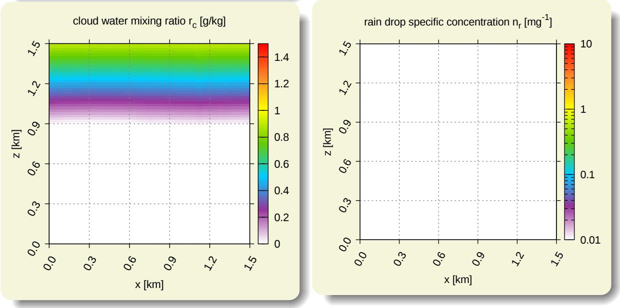


Lagrangian/Monte-Carlo scheme
(Shima et al. 2009)

2-moment bulk scheme
(Morrison & Grabowski 2007)

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

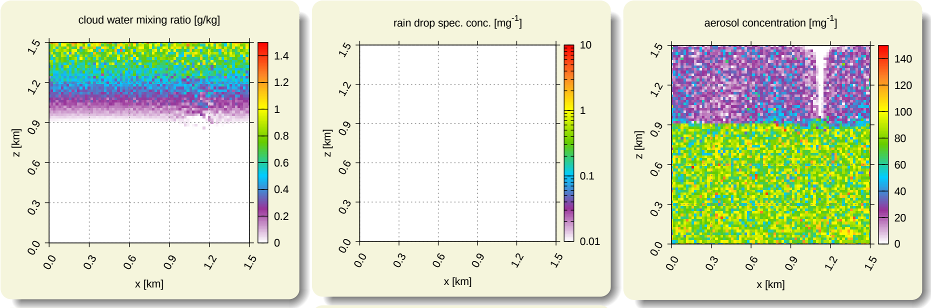


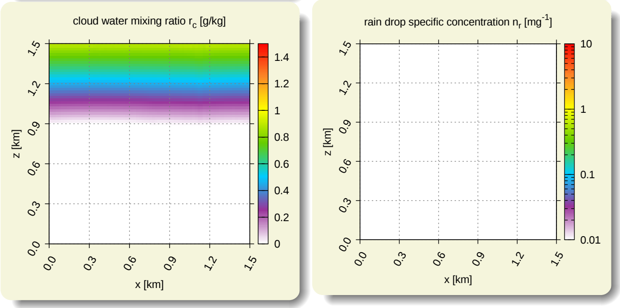


Lagrangian/Monte-Carlo scheme (Shima et al. 2009)

2-moment bulk scheme (Morrison & Grabowski 2007)

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



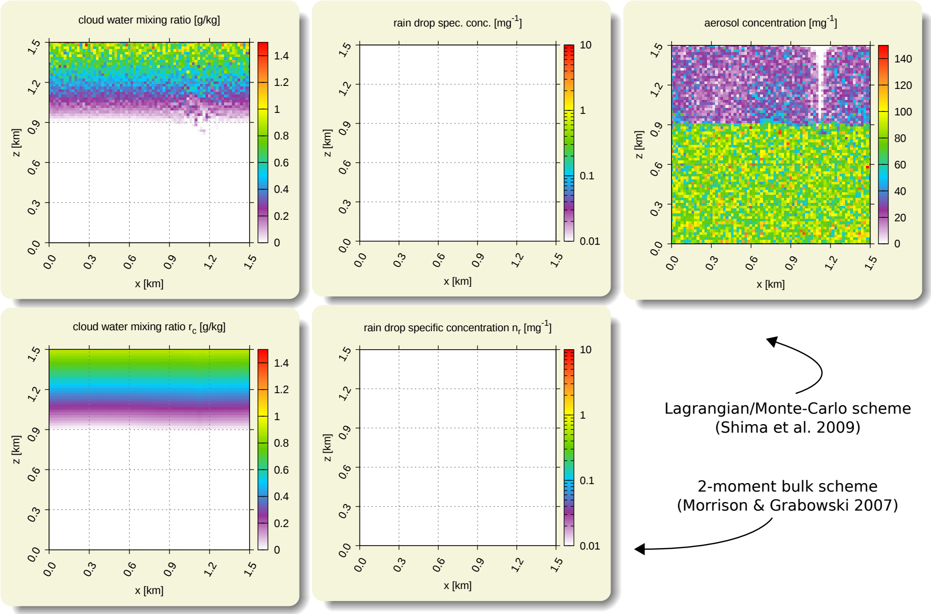


Lagrangian/Monte-Carlo scheme
(Shima et al. 2009)

2-moment bulk scheme
(Morrison & Grabowski 2007)

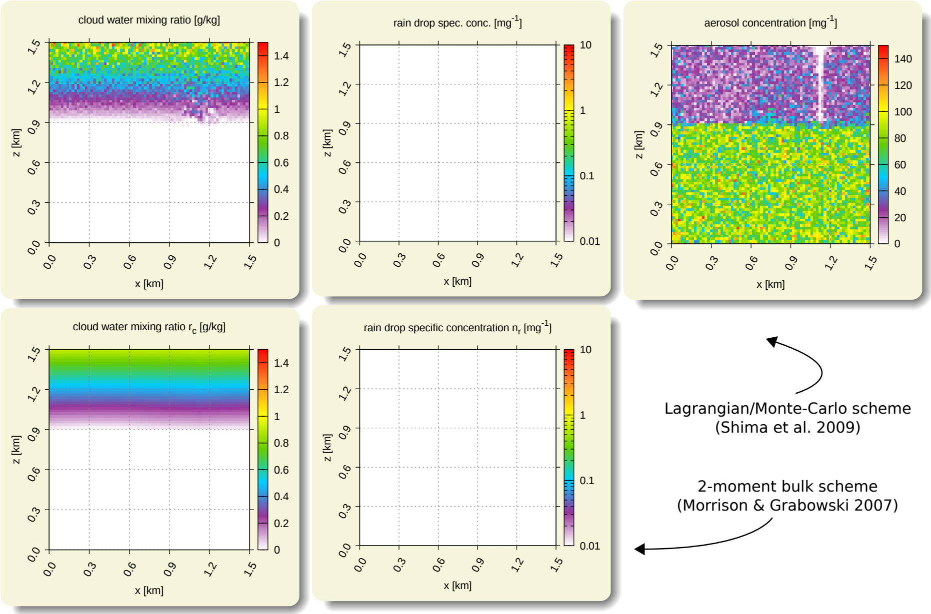
libcloudph++: VOCALS-inspired aerosol processing set-up

XXXXXXXXXXXXXXXX

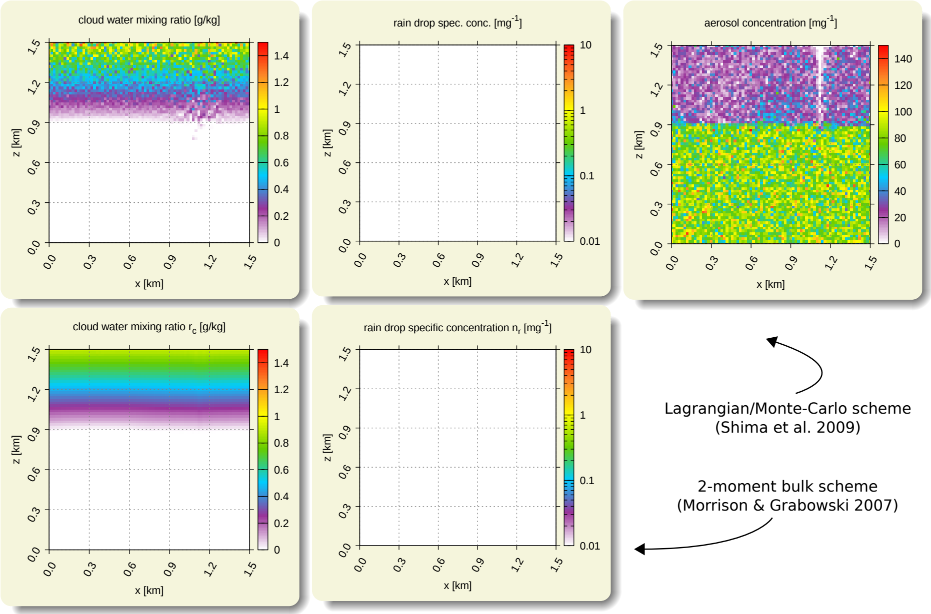


libcloudph++: VOCALS-inspired aerosol processing set-up

xxxxxxxxxxxxxxxxx

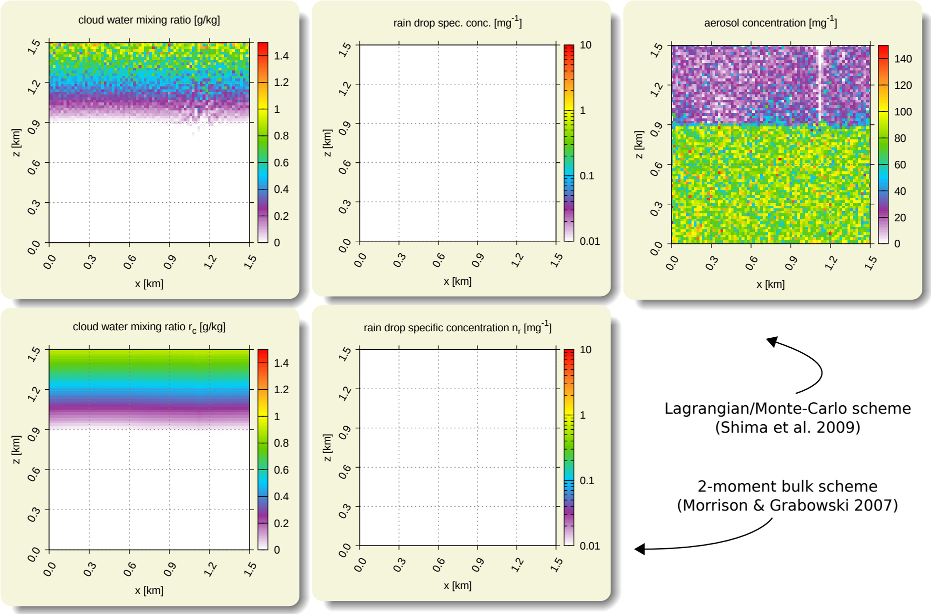


libcloudph++: VOCALS-inspired aerosol processing set-up



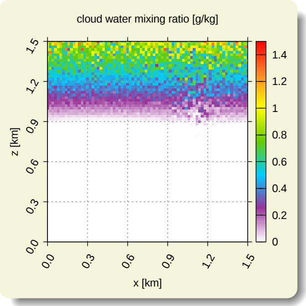
libcloudph++: VOCALS-inspired aerosol processing set-up

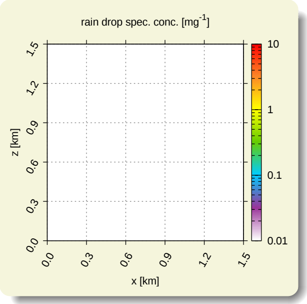
XXXXXXXXXXXXXXXXXXXX

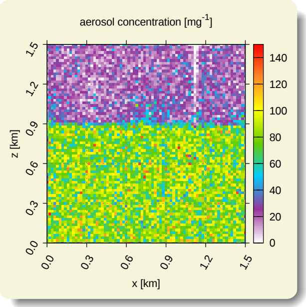


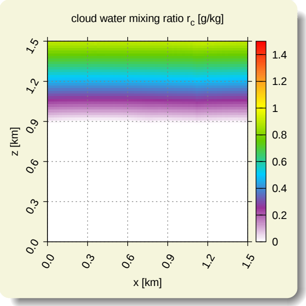
libcloudph++: VOCALS-inspired aerosol processing set-up

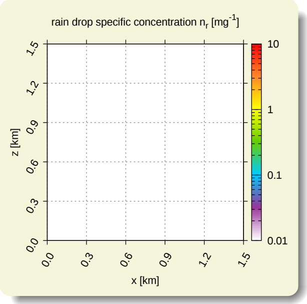
xxxxxxxxxxxxxxxxxxxxx







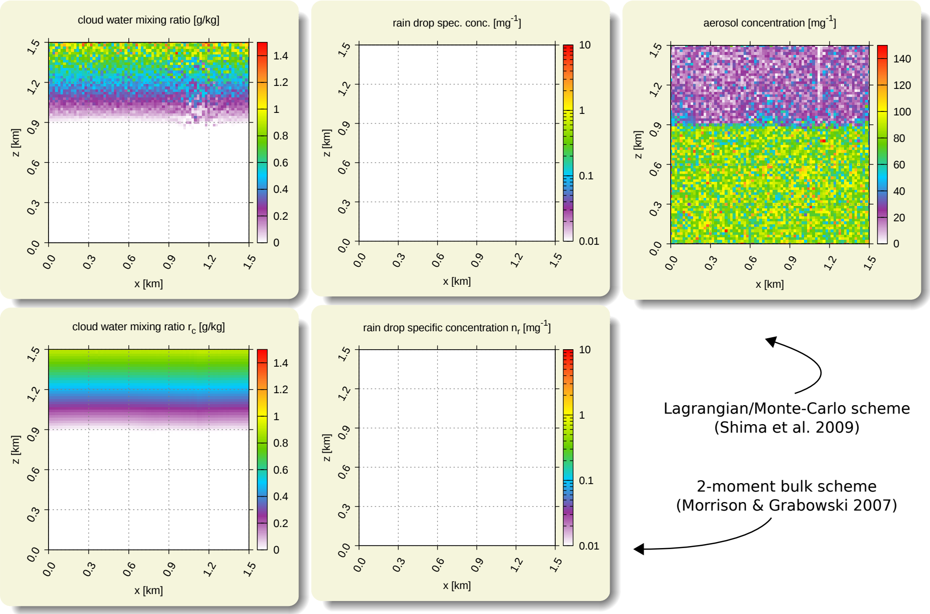




Lagrangian/Monte-Carlo scheme
(Shima et al. 2009)

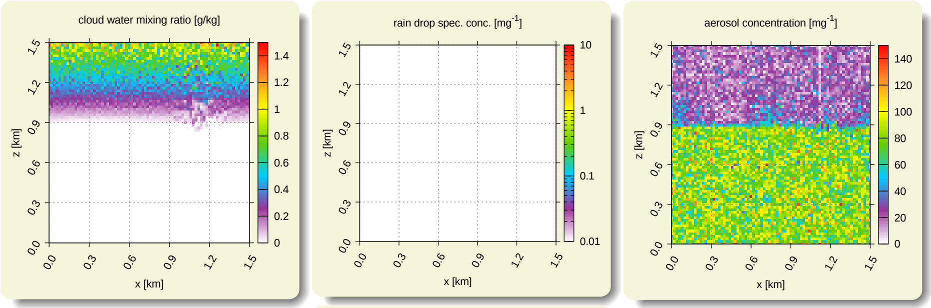
2-moment bulk scheme
(Morrison & Grabowski 2007)

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

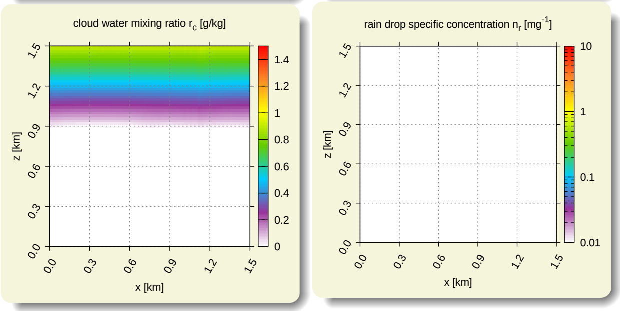


libcloudph++: VOCALS-inspired aerosol processing set-up

xxxxxxxxxxxxxxxxxxxxxxx

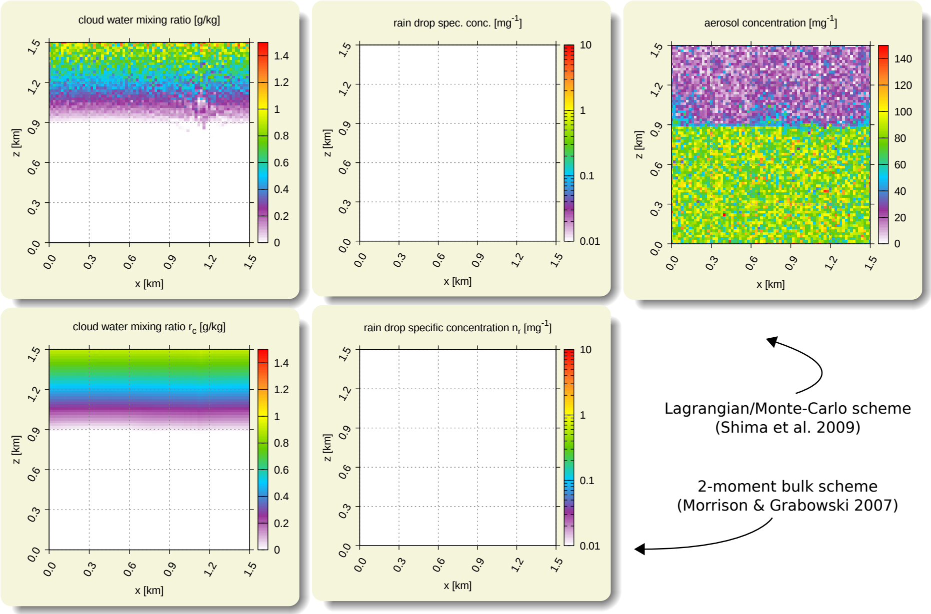


Lagrangian/Monte-Carlo scheme
(Shima et al. 2009)



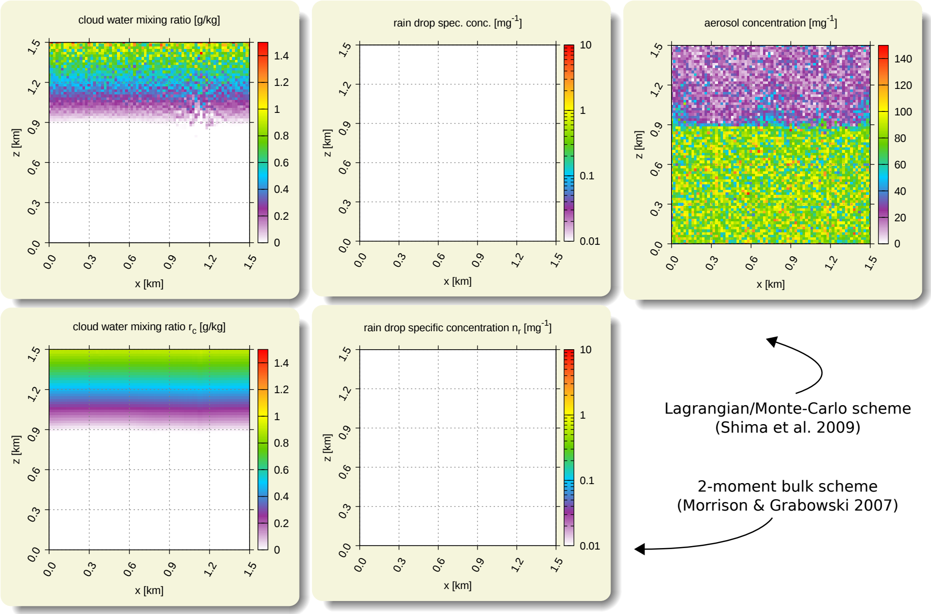
2-moment bulk scheme
(Morrison & Grabowski 2007)

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



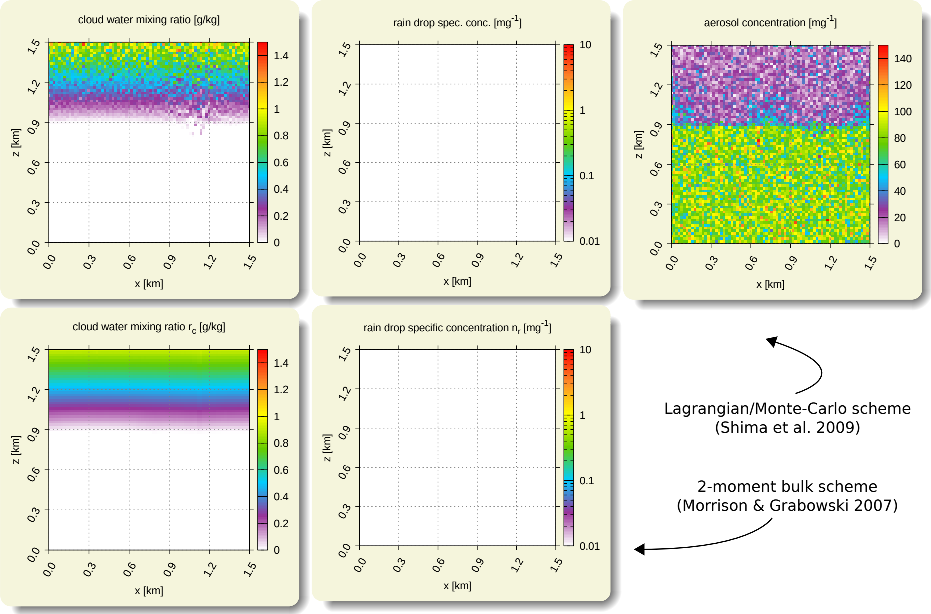
libcloudph++: VOCALS-inspired aerosol processing set-up

xxxxxxxxxxxxxxxxxxxxxxxxx



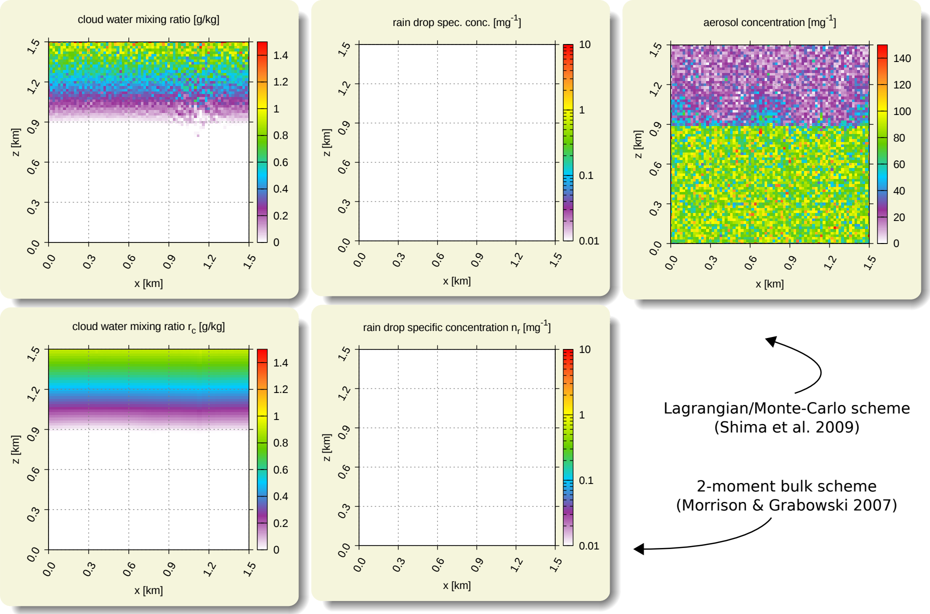
libcloudph++: VOCALS-inspired aerosol processing set-up

XXXXXXXXXXXXXXXXXXXXXXXXX

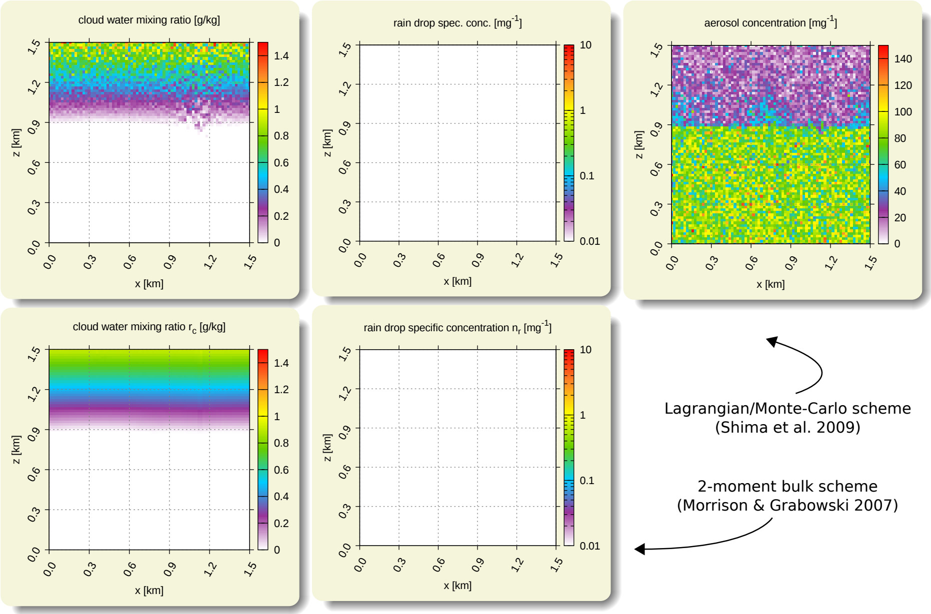


libcloudph++: VOCALS-inspired aerosol processing set-up

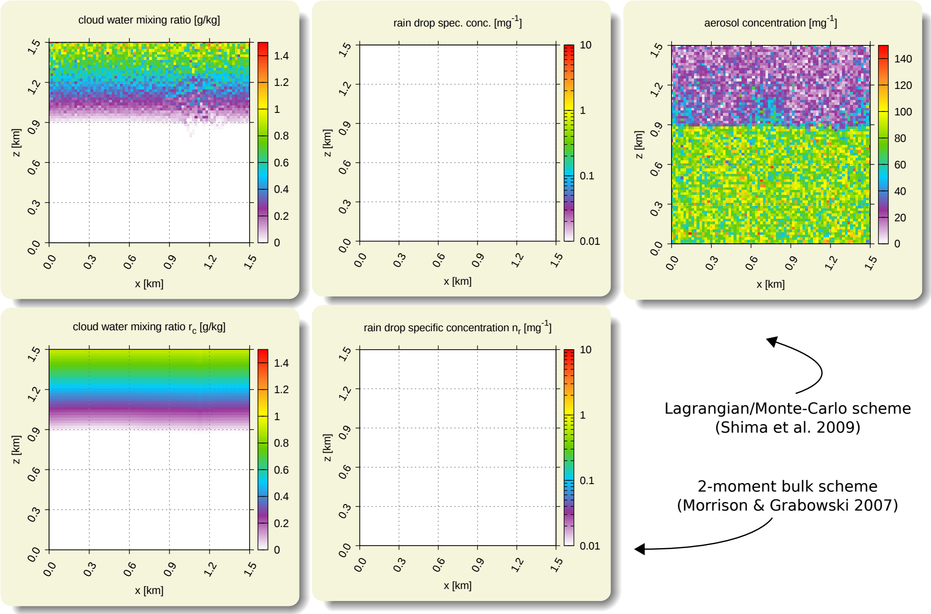
xxxxxxxxxxxxxxxxxxxxxxxxxx



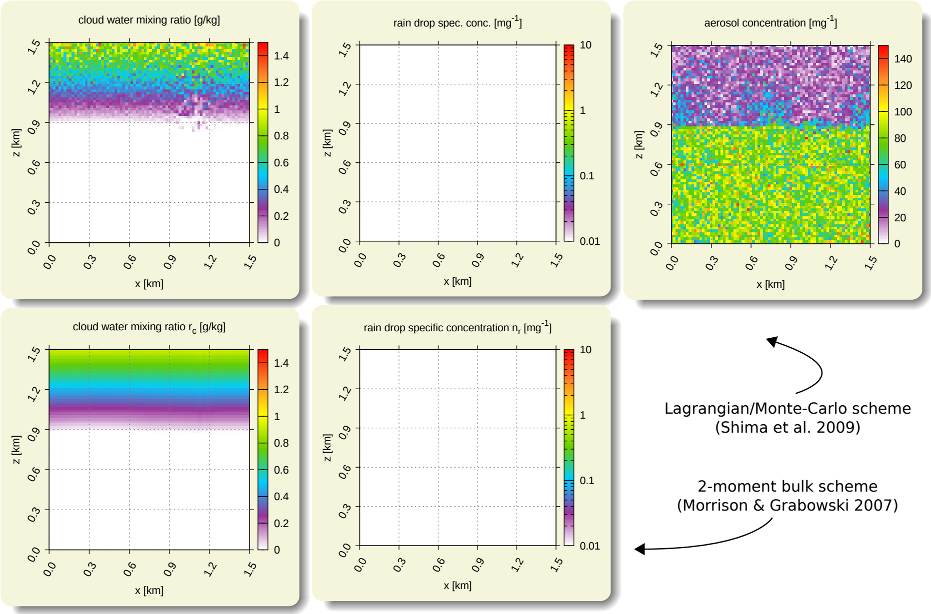
libcloudph++: VOCALS-inspired aerosol processing set-up



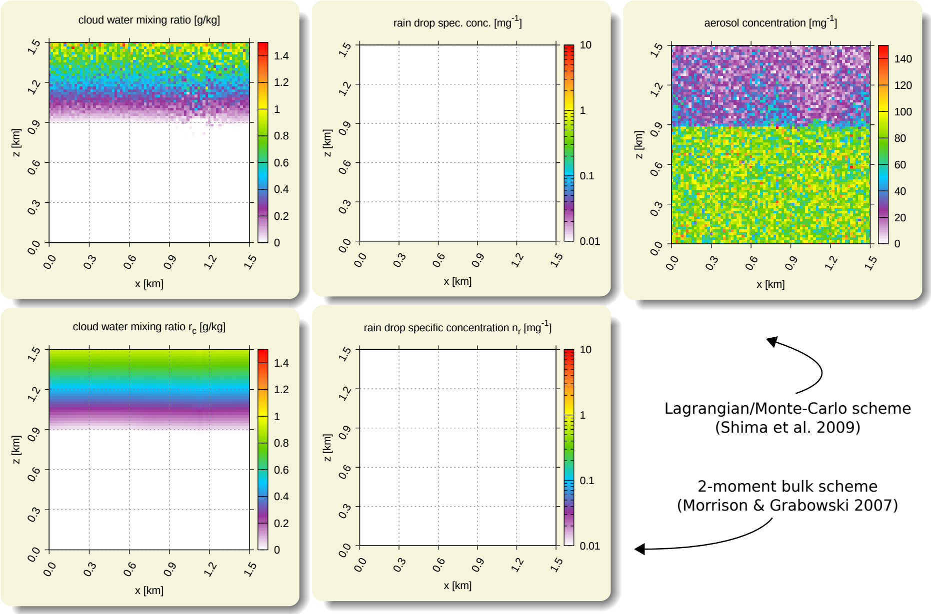
libcloudph++: VOCALS-inspired aerosol processing set-up



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



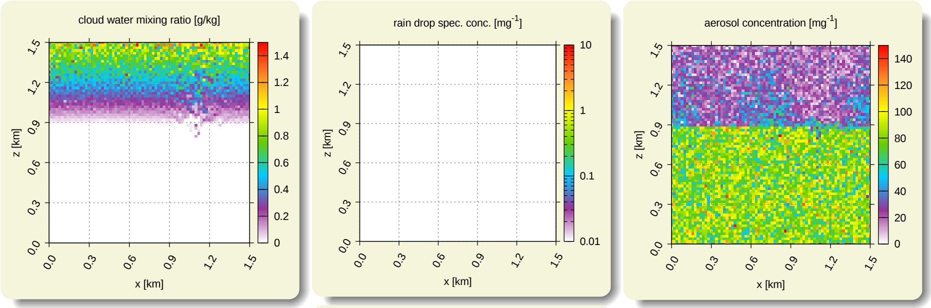
libcloudph++: VOCALS-inspired aerosol processing set-up



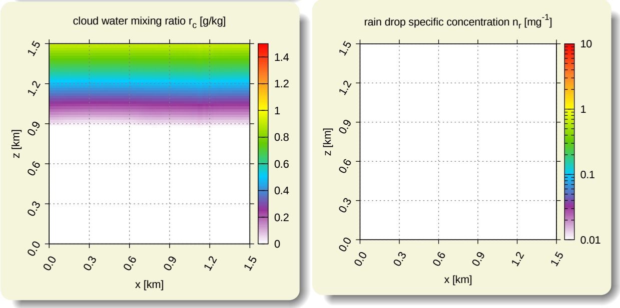
Lagrangian/Monte-Carlo scheme
(Shima et al. 2009)

2-moment bulk scheme
(Morrison & Grabowski 2007)

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



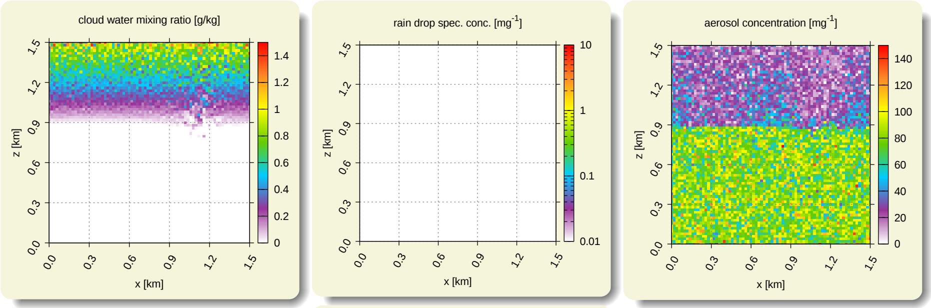
Lagrangian/Monte-Carlo scheme
(Shima et al. 2009)

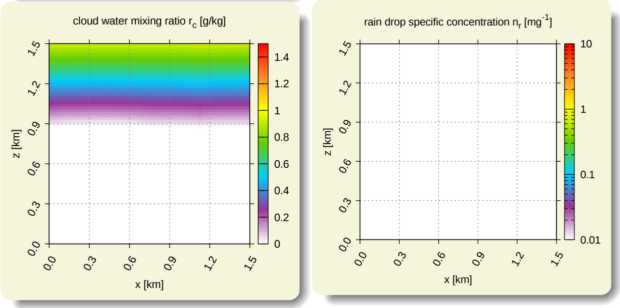


2-moment bulk scheme
(Morrison & Grabowski 2007)

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

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

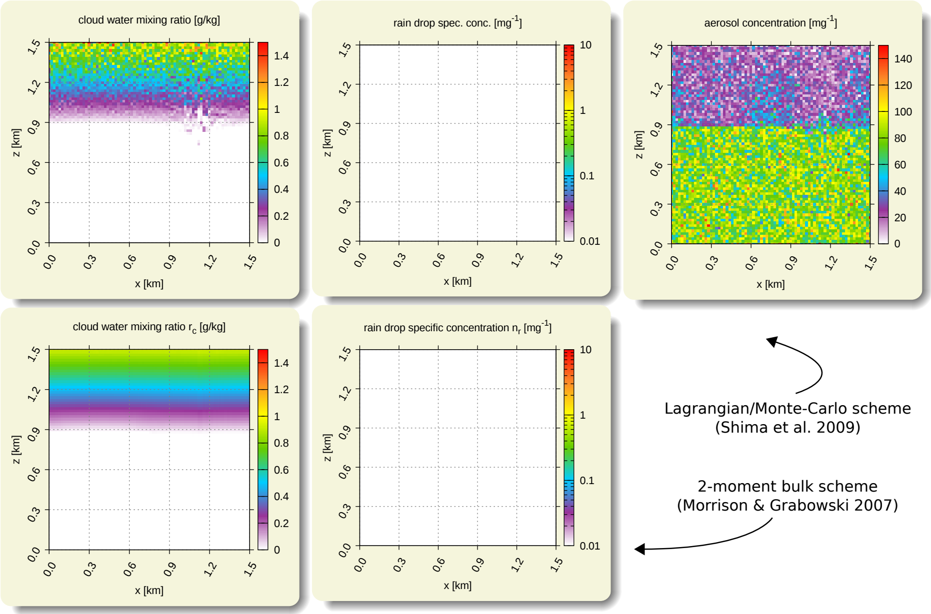




Lagrangian/Monte-Carlo scheme (Shima et al. 2009)

2-moment bulk scheme (Morrison & Grabowski 2007)

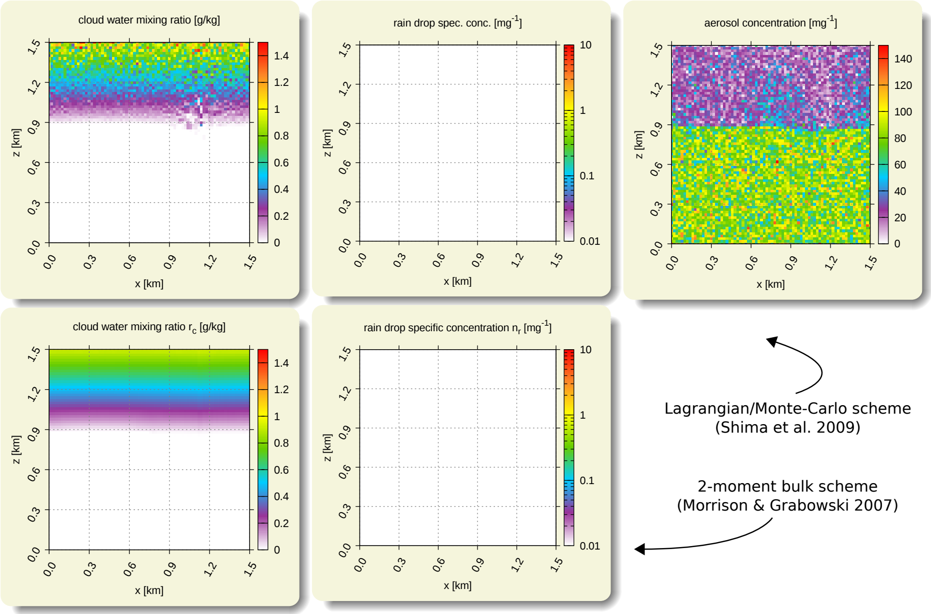
libcloudph++: VOCALS-inspired aerosol processing set-up



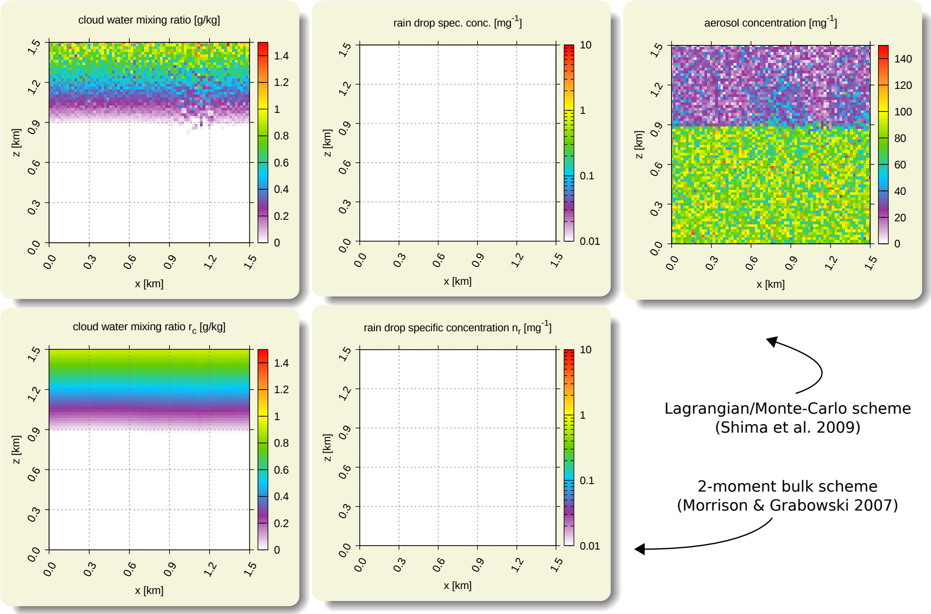
Lagrangian/Monte-Carlo scheme
(Shima et al. 2009)

2-moment bulk scheme
(Morrison & Grabowski 2007)

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

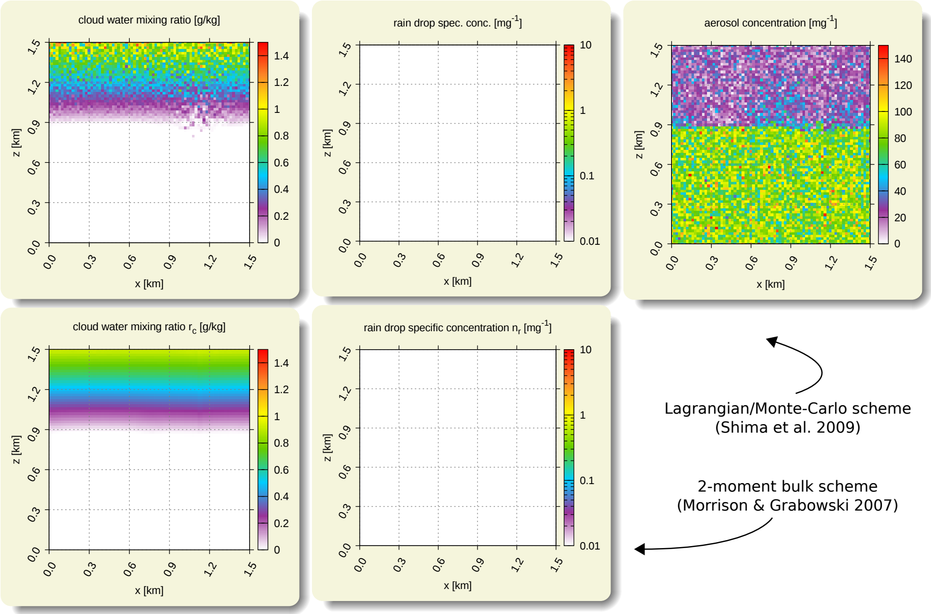


libcloudph++: VOCALS-inspired aerosol processing set-up

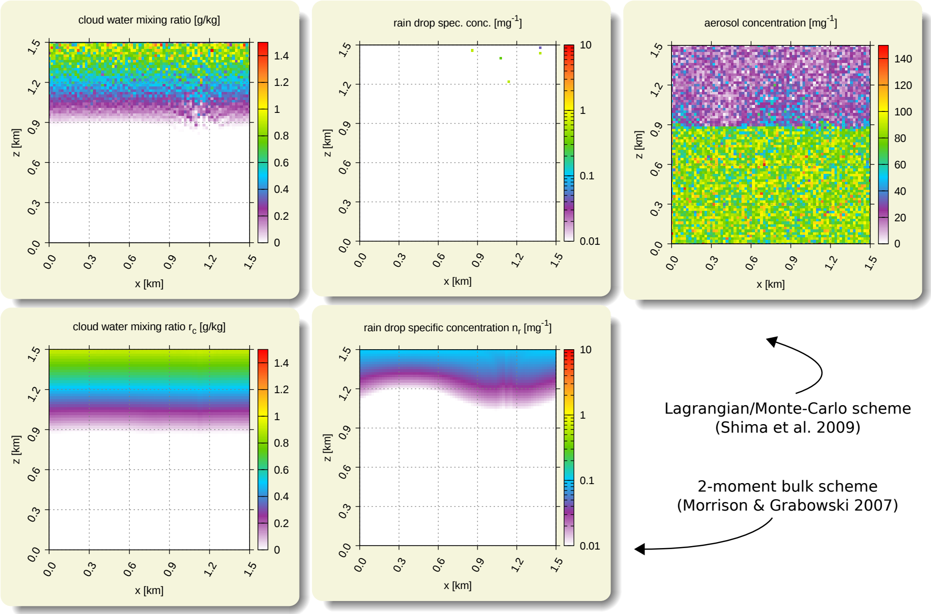


libcloudph++: VOCALS-inspired aerosol processing set-up

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx



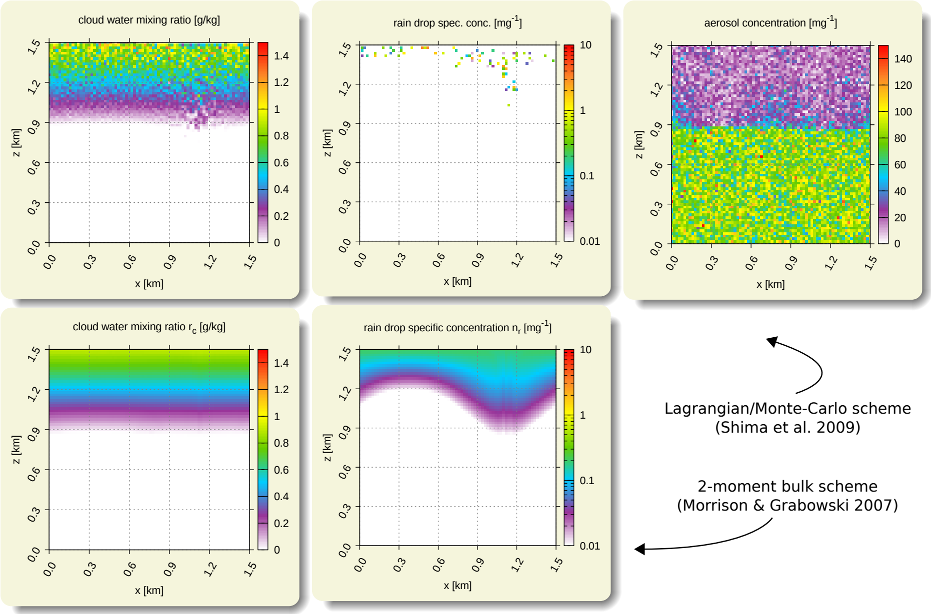
libcloudph++: VOCALS-inspired aerosol processing set-up



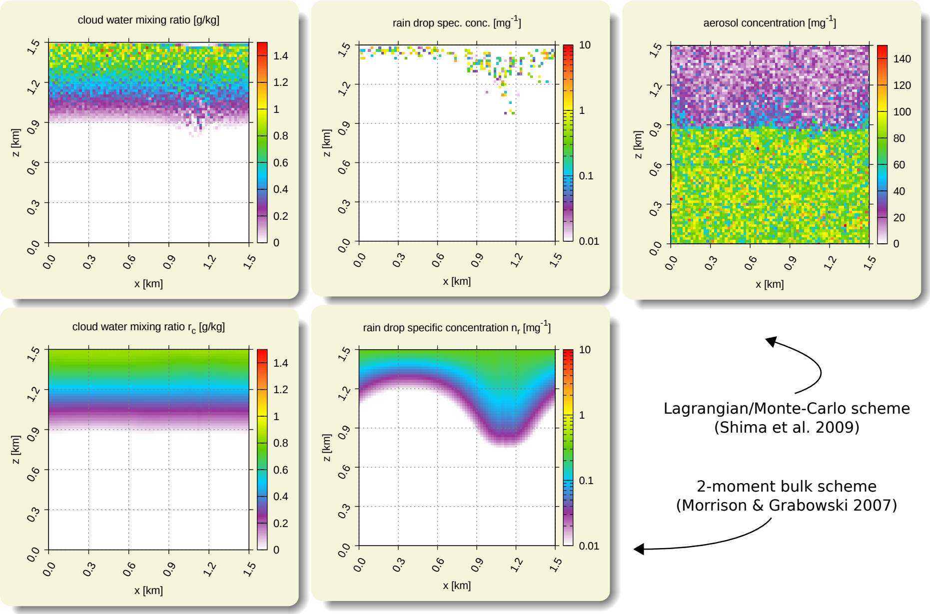
Lagrangian/Monte-Carlo scheme (Shima et al. 2009)

2-moment bulk scheme (Morrison & Grabowski 2007)

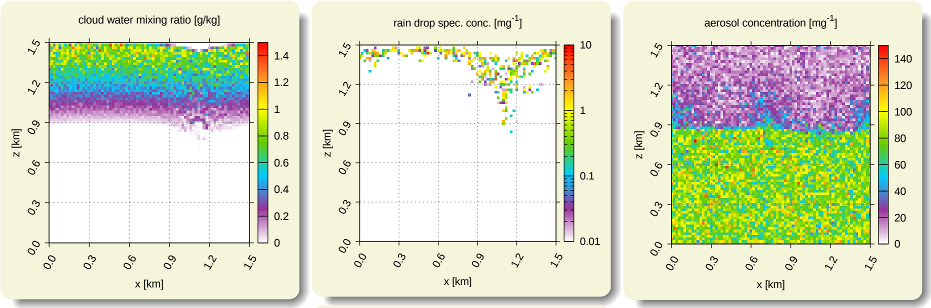
libcloudph++: VOCALS-inspired aerosol processing set-up

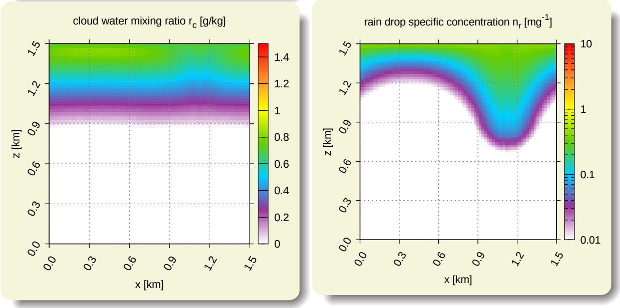


libcloudph++: VOCALS-inspired aerosol processing set-up



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

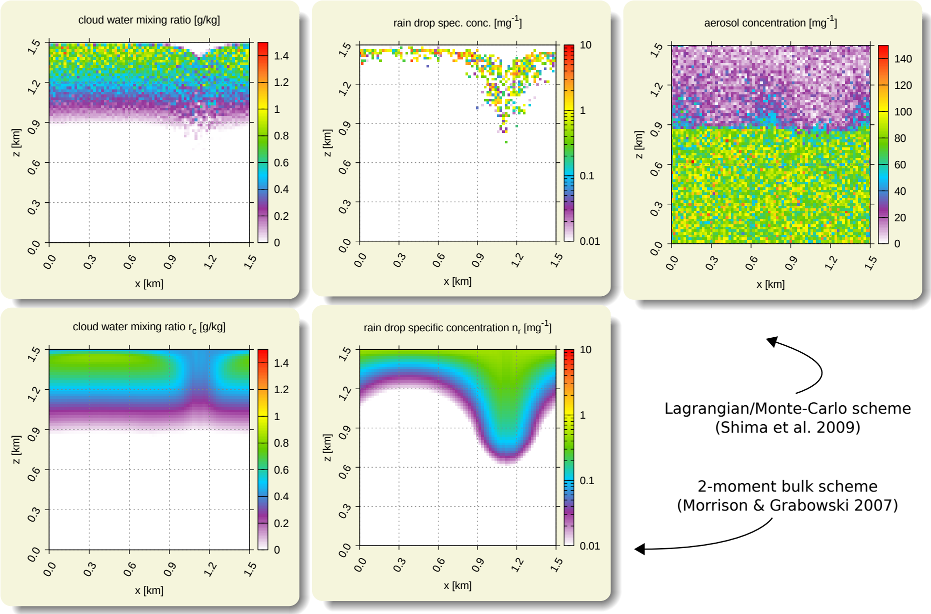




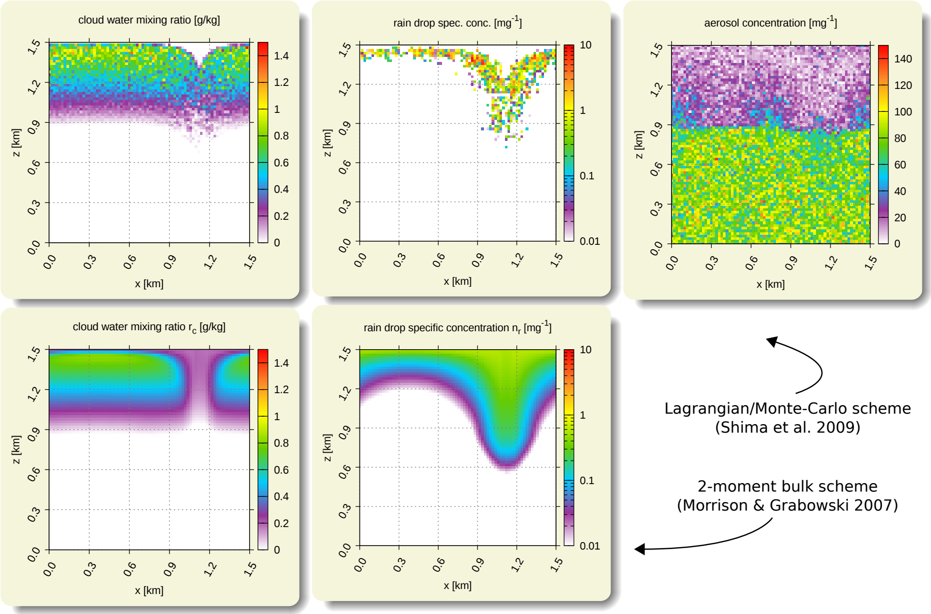
Lagrangian/Monte-Carlo scheme
(Shima et al. 2009)

2-moment bulk scheme
(Morrison & Grabowski 2007)

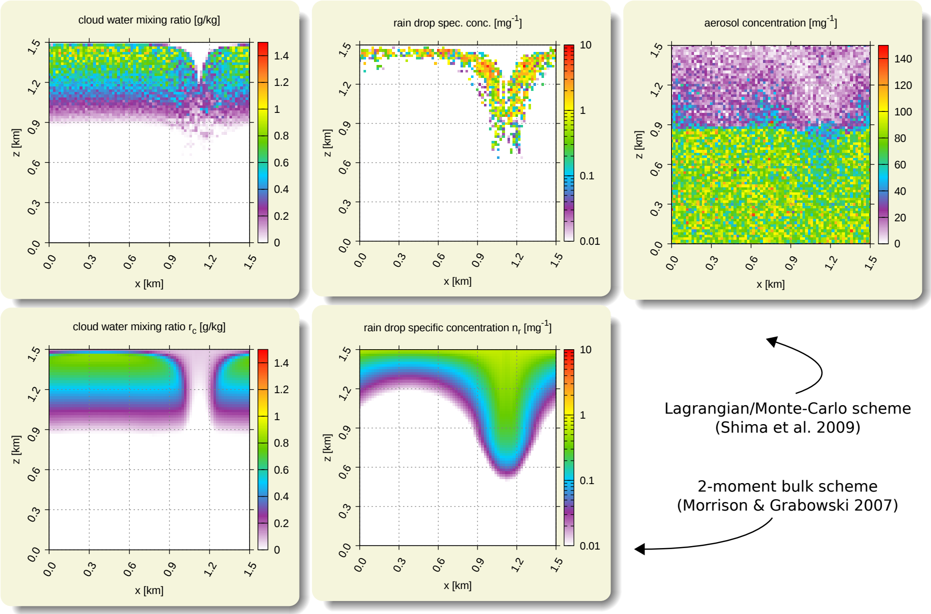
libcloudph++: VOCALS-inspired aerosol processing set-up



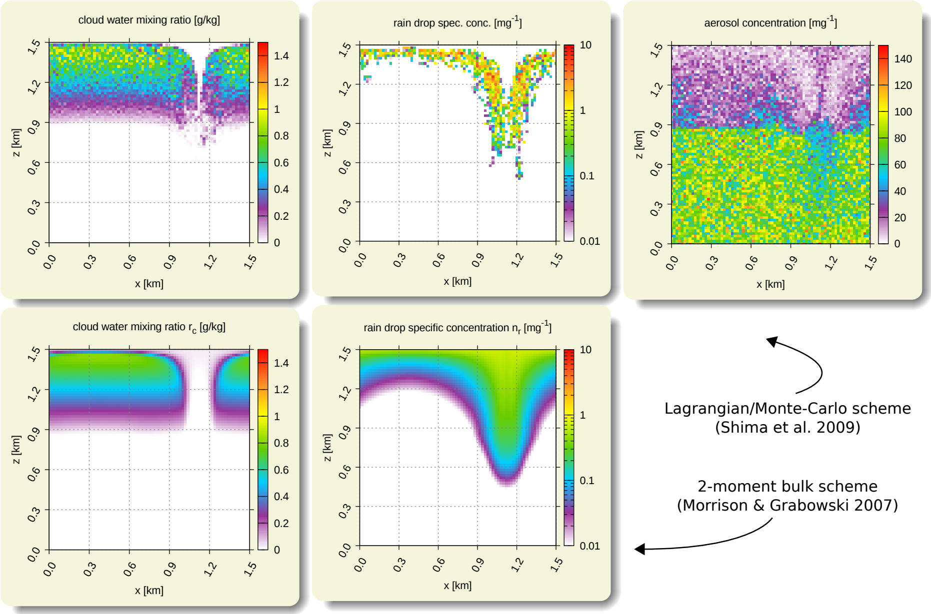
libcloudph++: VOCALS-inspired aerosol processing set-up



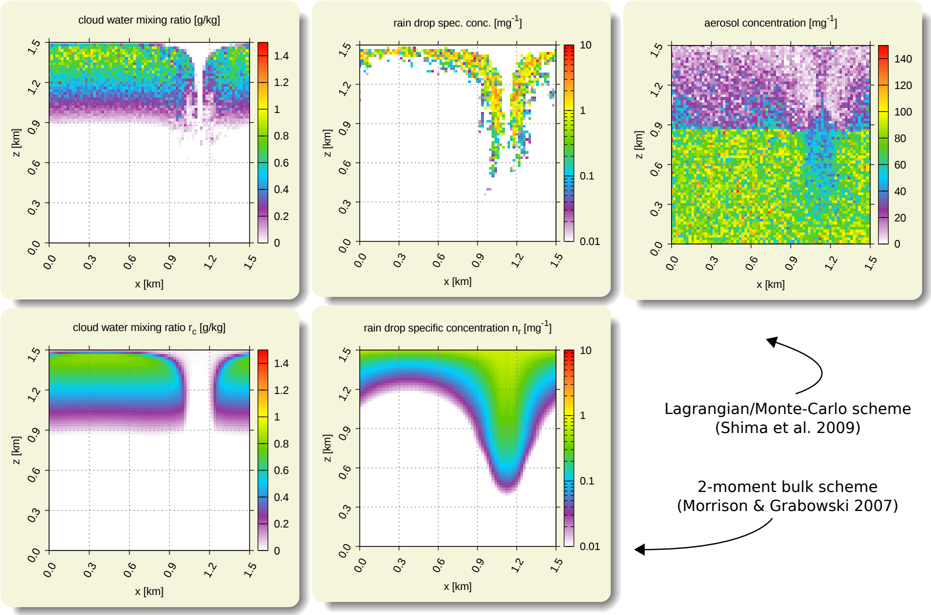
libcloudph++: VOCALS-inspired aerosol processing set-up



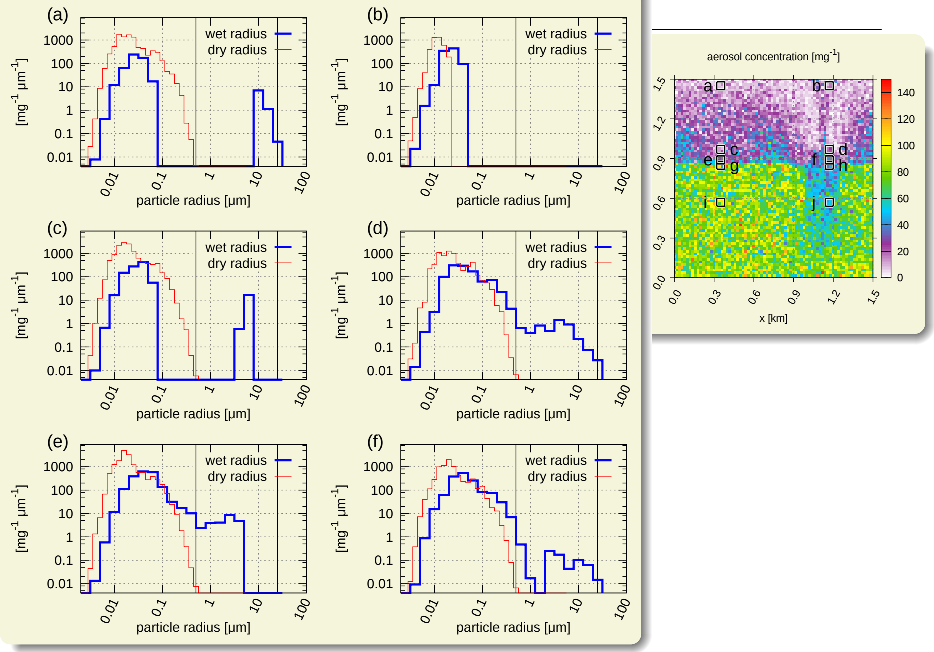
libcloudph++: VOCALS-inspired aerosol processing set-up



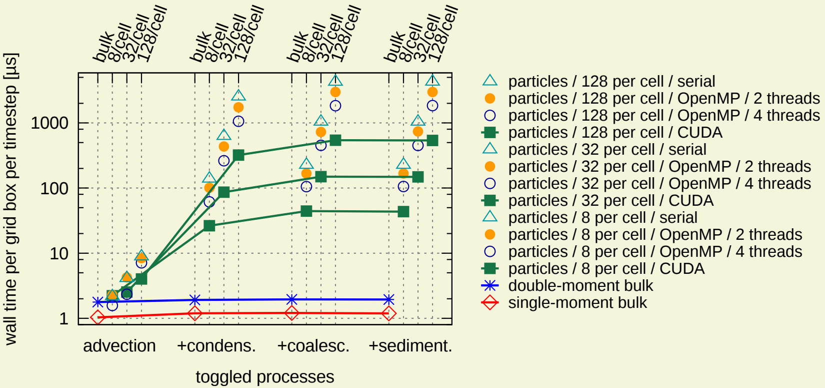
libcloudph++: VOCALS-inspired aerosol processing set-up



2×2 cell particle-derived spectra



libcloudph++: GPU-resident option for Lagrangian scheme



implemented using Thrust: OpenMP/GPU choice with no code modif.

libcloudph++: summary & some technicalities

free & open source C++ library of algorithms for cloud μ-physics

libcloudph++: summary & some technicalities

free & open source C++ library of algorithms for cloud μ-physics

key features:

- reusable – API documented in the paper

libcloudph++: summary & some technicalities

free & open source C++ library of algorithms for cloud μ-physics

key features:

- reusable – API documented in the paper
- three schemes (all written from scratch):
  - 1-moment: saturation adjustment / Kessler
  - 2-moment: with N_c and S prediction (Morrison-Grabowski '08)
  - Lagrangian with Monte-Carlo coalescence (Shima et al. 2009)

libcloudph++: summary & some technicalities

free & open source C++ library of algorithms for cloud μ-physics

key features:

- reusable – API documented in the paper
- three schemes (all written from scratch):
  - 1-moment: saturation adjustment / Kessler
  - 2-moment: with N_c and S prediction (Morrison-Grabowski '08)
  - Lagrangian with Monte-Carlo coalescence (Shima et al. 2009)
- Lagrangian scheme optionally GPU-resident

libcloudph++: summary & some technicalities

free & open source C++ library of algorithms for cloud μ-physics

key features:

- reusable – API documented in the paper
- three schemes (all written from scratch):
  - 1-moment: saturation adjustment / Kessler
  - 2-moment: with N_c and S prediction (Morrison-Grabowski '08)
  - Lagrangian with Monte-Carlo coalescence (Shima et al. 2009)
- Lagrangian scheme optionally GPU-resident
- compact code (500 / 1000 / 4500 LOC)

libcloudph++: summary & some technicalities

free & open source C++ library of algorithms for cloud μ-physics

key features:

- reusable – API documented in the paper
- three schemes (all written from scratch):
  - 1-moment: saturation adjustment / Kessler
  - 2-moment: with N_c and S prediction (Morrison-Grabowski '08)
  - Lagrangian with Monte-Carlo coalescence (Shima et al. 2009)
- Lagrangian scheme optionally GPU-resident
- compact code (500 / 1000 / 4500 LOC)
- written using Boost.units – compile-time dimensional analysis

libcloudph++: ongoing work

libmpdata++

libcloudph++

libcloudph++: ongoing work

libmpdata++

- higher-order operators for subgrid-scale modelling
  (with Piotr Smolarkiewicz & Maciej Waruszewski / U. Warsaw)

libcloudph++

libcloudph++: ongoing work

libmpdata++

- higher-order operators for subgrid-scale modelling
  (with Piotr Smolarkiewicz & Maciej Waruszewski / U. Warsaw)
- distributed memory parallelisation (Boost.MPI)

libcloudph++

libcloudph++: ongoing work

libmpdata++

- higher-order operators for subgrid-scale modelling
  (with Piotr Smolarkiewicz & Maciej Waruszewski / U. Warsaw)
- distributed memory parallelisation (Boost.MPI)

libcloudph++

- aqueous chemistry for the Lagrangian scheme
  (with Anna Jaruga / U. Warsaw)

libcloudph++: ongoing work

libmpdata++

- higher-order operators for subgrid-scale modelling
  (with Piotr Smolarkiewicz & Maciej Waruszewski / U. Warsaw)
- distributed memory parallelisation (Boost.MPI)

libcloudph++

- aqueous chemistry for the Lagrangian scheme
  (with Anna Jaruga / U. Warsaw)
- drop breakup & adaptive timesteps in the Lagrangian scheme
  (with Shin-ichiro Shima / U. Hyogo)

libcloudph++: ongoing work

libmpdata++

- higher-order operators for subgrid-scale modelling
  (with Piotr Smolarkiewicz & Maciej Waruszewski / U. Warsaw)
- distributed memory parallelisation (Boost.MPI)

libcloudph++

- aqueous chemistry for the Lagrangian scheme
  (with Anna Jaruga / U. Warsaw)
- drop breakup & adaptive timesteps in the Lagrangian scheme
  (with Shin-ichiro Shima / U. Hyogo)
- Python bindings (Boost.Python)
  (with Dorota Jarecka / NCAR)

libcloudph++: ongoing work

libmpdata++

- higher-order operators for subgrid-scale modelling
  (with Piotr Smolarkiewicz & Maciej Waruszewski / U. Warsaw)
- distributed memory parallelisation (Boost.MPI)

libcloudph++

- aqueous chemistry for the Lagrangian scheme
  (with Anna Jaruga / U. Warsaw)
- drop breakup & adaptive timesteps in the Lagrangian scheme
  (with Shin-ichiro Shima / U. Hyogo)
- Python bindings (Boost.Python)
  (with Dorota Jarecka / NCAR)
- DALES/libcloudph++ coupling
  (with Harm Jonker / TU Delft)

Thank you for your attention!

Thank you for your attention!

- postdoc position at our group: http://foss.igf.fuw.edu.pl/

Thank you for your attention!

- postdoc position at our group: http://foss.igf.fuw.edu.pl/
- libmpdata++ paper: http://arxiv.org/abs/1407.1309
- libcloudph++ paper: http://arxiv.org/abs/1310.1905

Thank you for your attention!

- postdoc position at our group: http://foss.igf.fuw.edu.pl/
- libmpdata++ paper: http://arxiv.org/abs/1407.1309
- libcloudph++ paper: http://arxiv.org/abs/1310.1905
- code repositories: http://github.com/igfuw/

Thank you for your attention!

- postdoc position at our group: http://foss.igf.fuw.edu.pl/
- libmpdata++ paper: http://arxiv.org/abs/1407.1309
- libcloudph++ paper: http://arxiv.org/abs/1310.1905
- code repositories: http://github.com/igfuw/

acknowledgements:

- My visit to NCAR is supported by Foundation for Polish Science (Mentoring programme)

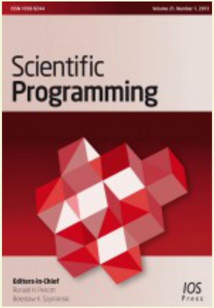
Thank you for your attention!

- postdoc position at our group: http://foss.igf.fuw.edu.pl/
- libmpdata++ paper: http://arxiv.org/abs/1407.1309
- libcloudph++ paper: http://arxiv.org/abs/1310.1905
- code repositories: http://github.com/igfuw/

acknowledgements:

- My visit to NCAR is supported by Foundation for Polish Science (Mentoring programme)
- Development of libmpdata++ and libcloudph++ have been supported by Poland's National Science Centre (decision no. 2012/06/M/ST10/00434)

MPDATA in C++, Fortran and Python



Formula translation in Blitz++, NumPy and modern Fortran: A case study of the language choice tradeoffs

Sylwester Arabas[1], Dorota Jarecka[1], Anna Jaruga[1], Maciej Fijałkowski[2]

[1]Institute of Geophysics, Faculty of Physics, University of Warsaw
[2]PyPy Team

Journal	Scientific Programming
DOI	10.3233/SPR-140379
Online Date	Monday, March 24, 2014