TECHNISCHE UNIVERSITÄT DRESDEN

Center for Information Services and High Performance Computing (ZIH)

FD4: A Framework for Highly Scalable Load Balancing and Coupling of Multiphase Models

ICNAAM 2010, September 20, Rhodes, Greece

Matthias Lieber [a], Verena Grützun [b], Ralf Wolke [c], Matthias S. Müller [a], Wolfgang E. Nagel [a]

[a] Center for Information Services and High Performance Computing (ZIH), TU Dresden, Germany

[b] Max Planck Institute for Meteorology (MPI-M), Hamburg, Germany

[c] Leibniz Institute for Tropospheric Research (IfT), Leipzig, Germany

LEIBNIZ INSTITUTE FOR TROPOSPHERIC RESEARCH

Max-Planck-Institut für Meteorologie

Center for Information Services & High Performance Computing

Outline

- Introduction
  - Detailed Simulation of Clouds
  - Basic Idea
- Framework FD4
  - Key Features
  - More Details
- Application of FD4
  - Performance Comparison
- Conclusion & Outlook

Introduction: Detailed Simulation of Clouds

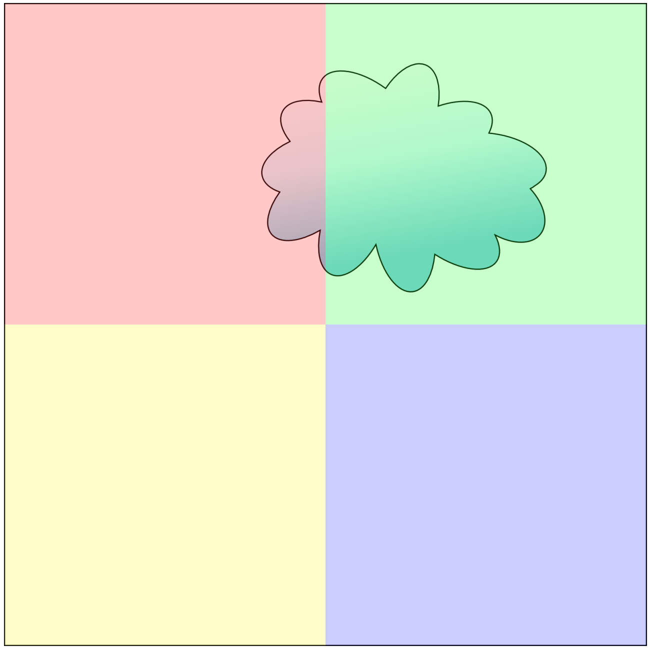
- Leibniz Institute for Tropospheric Research (IfT), Leipzig, Germany
- Goal: Detailed modeling of interactions between aerosol particles, clouds, and precipitation
- Development of the model system COSMO-SPECS, consisting of two coupled models:
  - COSMO Model: non-hydrostatic limited-area atmospheric model (www.cosmo-model.org)
  - SPECS: Cloud parameterization scheme of COSMO replaced by the detailed cloud model SPECS (SPECtral bin microphysicS) [Simmel06, Grützun08]

LEIBNIZ INSTITUTE FOR TROPOSPHERIC RESEARCH

DWD

Introduction: COSMO-SPECS Performance

- SPECS is very costly
  - > 99% of total runtime
- SPECS runtime varies strongly
  - Depending on range of droplet size distribution and the presence of frozen particles
- This leads to severe load imbalance
  - COSMO's parallelization is based on static 2D partitioning



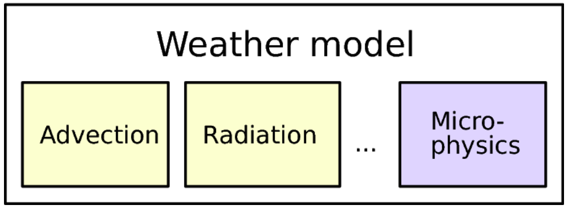
Dynamic load balancing needed to run realistic cases on large HPC systems

Introduction: Project Overview

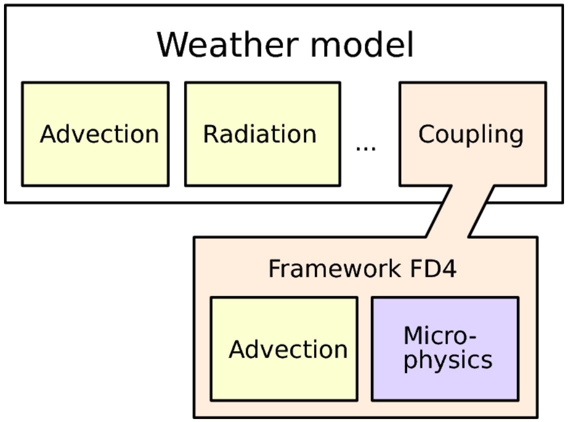
- *"Parallel coupling framework and advanced time integration methods for detailed cloud processes in atmospheric models"*
- Center for Information Services and High Performance Computing (ZIH), TU Dresden, Germany
  - Focus: "*Parallel coupling framework*"
  - Focus of this presentation
- Leibniz Institute for Tropospheric Research (IfT), Leipzig, Germany
  - Focus: "*Advanced time integration methods*"
  - Presentation yesterday by Martin Schlegel
- http://www.tu-dresden.de/zih/clouds

Introduction: Basic Idea

- Present approaches:
  - Cloud model is implemented as a sub-module within the weather model
  - Uses (static) data structures of the weather model

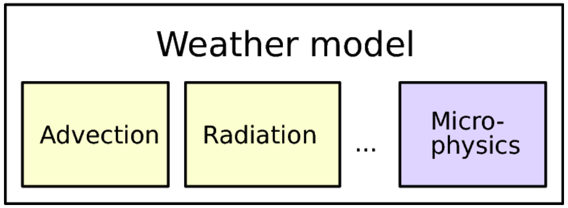


- Our idea:
  - Separate cloud model data from weather model data structures
  - Independent domain decompositions
  - Dynamic load balancing for the cloud model
  - (Re)couple weather and cloud model



Introduction: Basic Idea

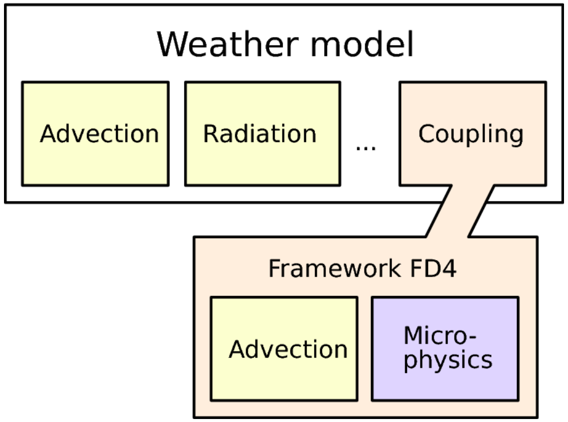
- Present approaches:
  - Cloud model is implemented as a sub-module within the weather model
  - Uses (static) data structures of the weather model



- Our idea:

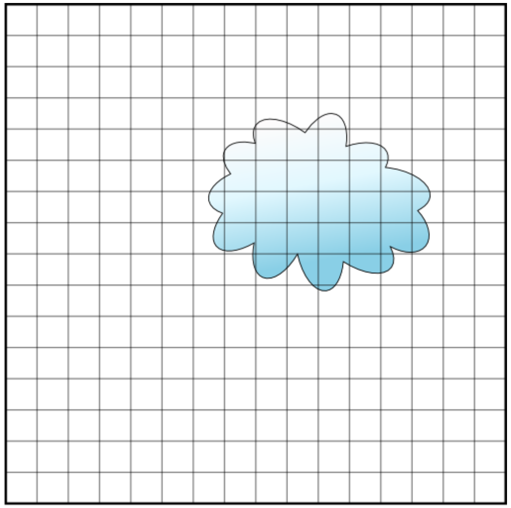
Functionality provided by FD4

  - Separate cloud model data from weather model data structures
  - Independent domain decompositions
  - Dynamic load balancing for the cloud model
  - (Re)couple weather and cloud model



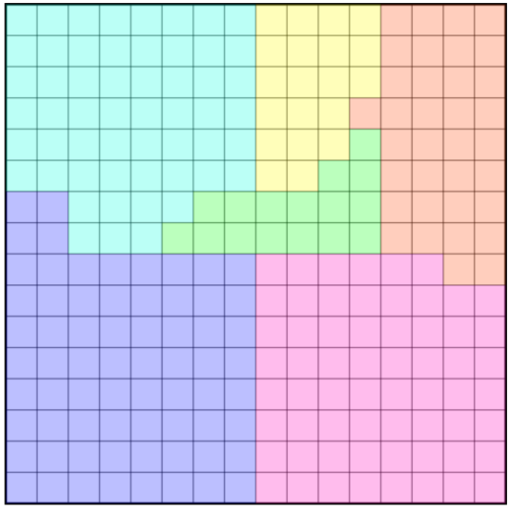
FD^4 = Four-Dimensional Distributed Dynamic Data structures

- Dynamic load balancing
  - Regular grid managed by FD4
  - Block-based 3D decomposition
- Model coupling
- Adaptive block mode
- 4th dimension



Framework FD4: Key Features

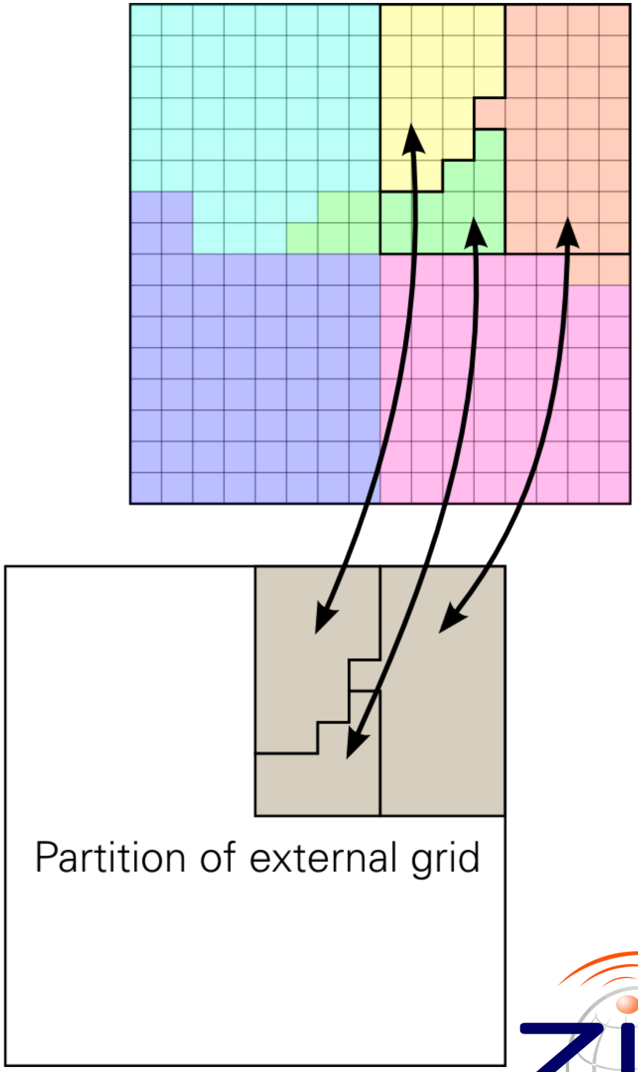
FD^4 = Four-Dimensional Distributed Dynamic Data structures



- Dynamic load balancing
  - Regular grid managed by FD4
  - Block-based 3D decomposition
- Model coupling
- Adaptive block mode
- 4th dimension

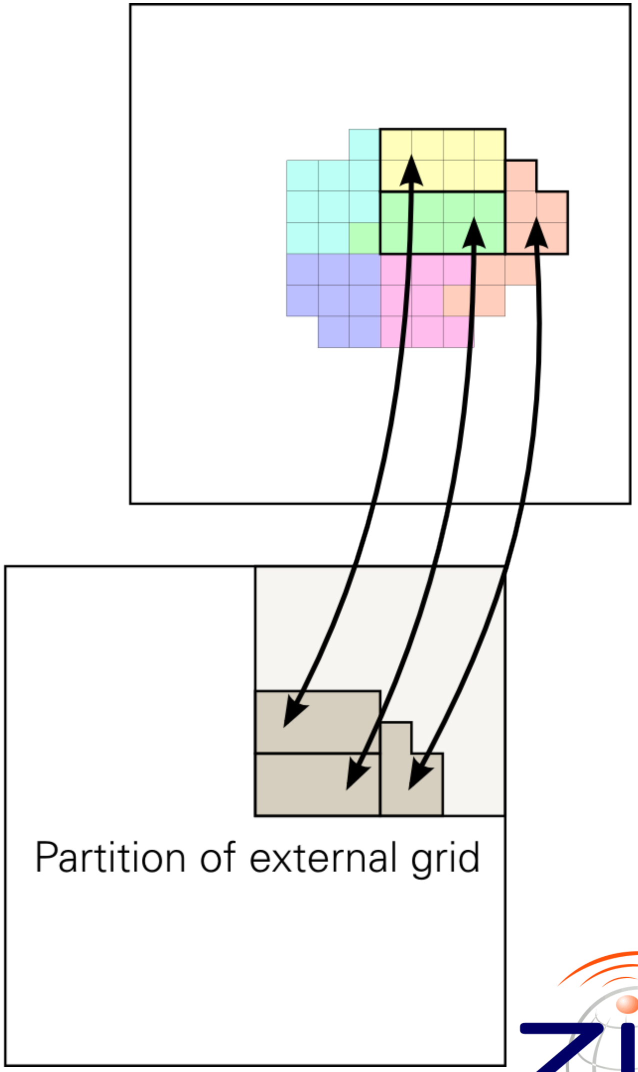
FD^4 = Four-Dimensional Distributed Dynamic Data structures

- Dynamic load balancing
- Model coupling
  - Data exchange between FD4 based model and external model
  - E.g. CFD or weather model
  - Direct data transfer between overlapping parts of of the partitions
- Adaptive block mode
- 4th dimension



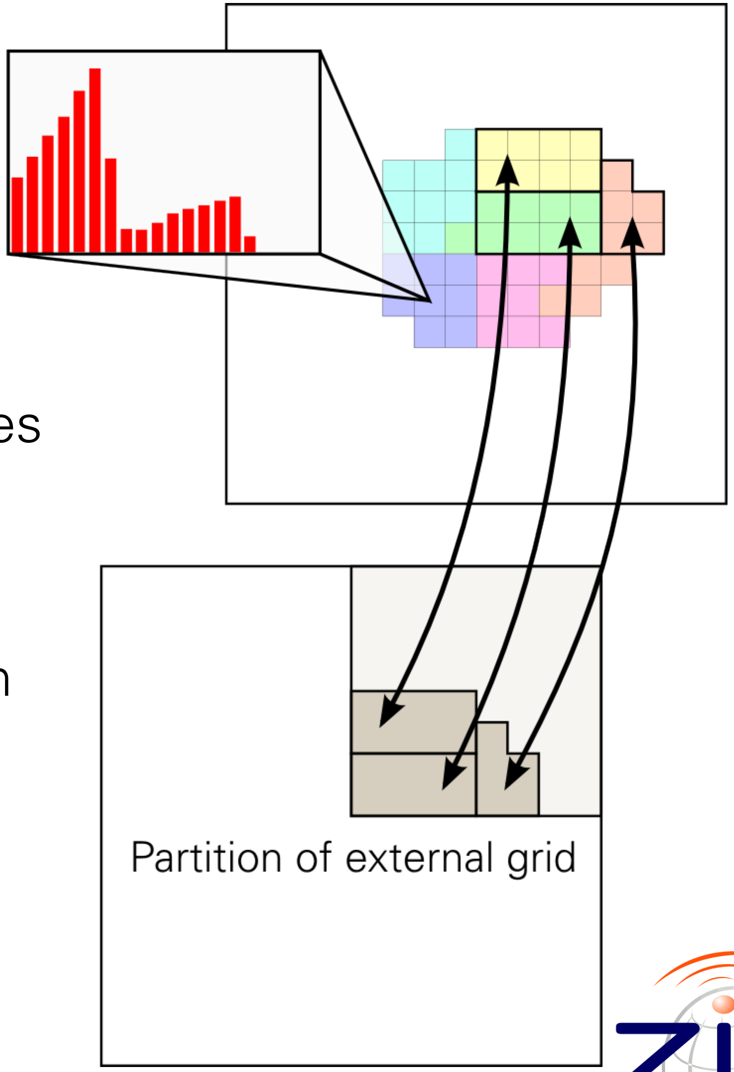
FD^4 = Four-Dimensional Distributed Dynamic Data structures

- Dynamic load balancing
- Model coupling
- Adaptive block mode
  - Save memory in case data and computations are required for a spatial subset only
  - Suitable for multiphase problems like drops, clouds, flame fronts
- 4th dimension



FD^4 = Four-Dimensional Distributed Dynamic Data structures

- Dynamic load balancing
- Model coupling
- Adaptive block mode
- 4th dimension
  - Extra dimension of grid variables
  - E.g. array of gas phase tracers, size resolving models
  - FD4 is optimized for a large 4th dimension
  - COSMO-SPECS requires 2 x 11 x 66 ~ 1500 values



Framework FD4: Software

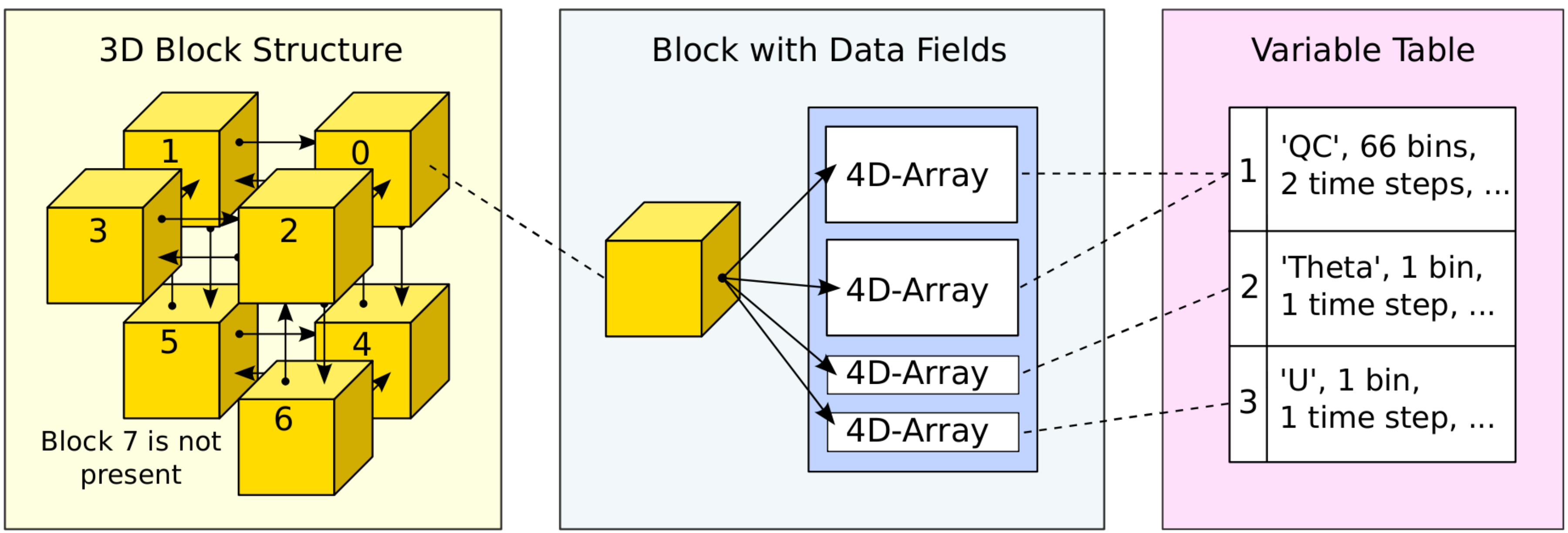
- FD4 is written in Fortran 95
- MPI-2 based parallelization
- (Simple) I/O interfaces to
  - NetCDF
  - Vis5D
- Open source software

```
! MPI initialization
call MPI_Init(err)
call MPI_Comm_rank(MPI_COMM_WORLD, rank, err)
call MPI_Comm_size(MPI_COMM_WORLD, nproc, err)
! create the domain and allocate memory
call fd4_domain_create(domain, nb, size,      &
     vartab, ng, peri, MPI_COMM_WORLD, err)
call fd4_util_allocate_all_blocks(domain, err)
! initialize ghost communication
call fd4_ghostcomm_create(ghostcomm, domain, &
     4, vars, steps, err)
! loop over time steps
do timestep=1,nsteps
  ! exchange ghosts
  call fd4_ghostcomm_exch(ghostcomm, err)
  ! loop over local blocks
  call fd4_iter_init(domain, iter)
  do while(associated(iter%cur))
    ! do some computations
    call compute_block(iter)
    call fd4_iter_next(iter)
  end do
  ! dynamic load balancing
  call fd4_balance_readjust(domain, err)
end do
```

Available at http://www.tu-dresden.de/zih/clouds

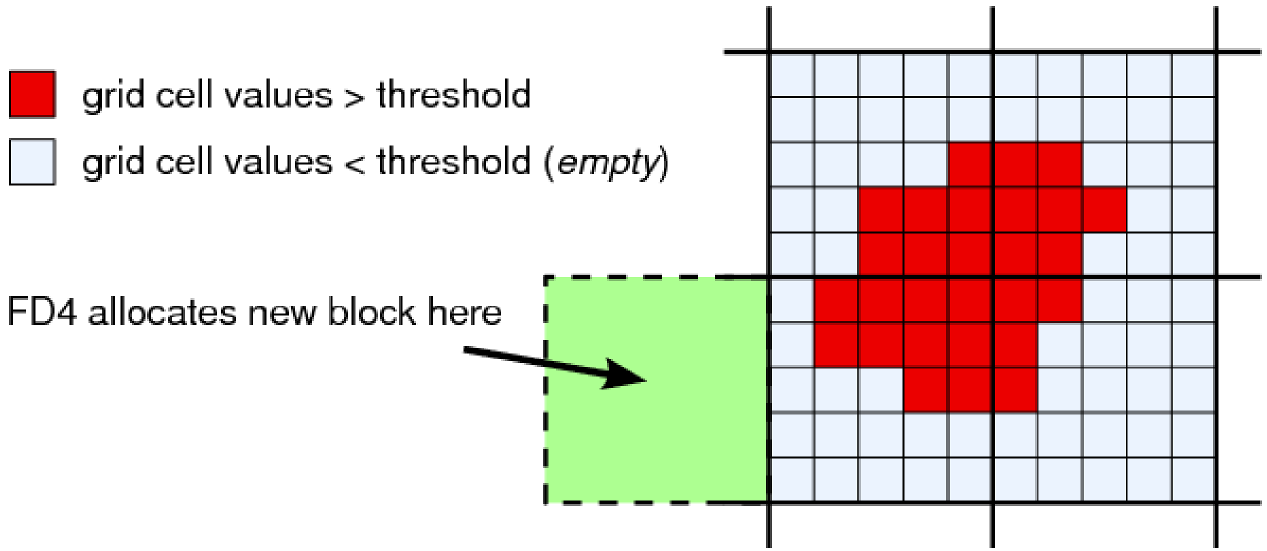
Framework FD4: Basic Data Structure

- 3D Regular grid is decomposed into blocks
- FD4 allocates data fields in the blocks based on variable table
  - Variable name
  - Length of 4th dimension
  - Number of time steps
  - Discretization, i.e. cell-centered or face-centered



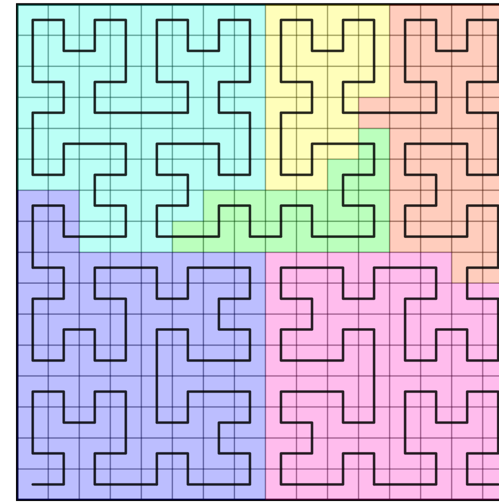
Framework FD4: Adaptive Block Mode

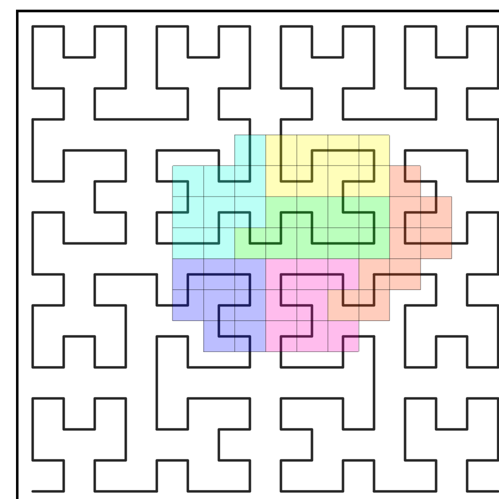
- Grid allocation adapts to spatial structure of simulated problem
- *Empty* blocks are not allocated
  - Defined by a threshold value for specified variables
  - I.e. block does not contain any quantities of a certain phase
- FD4 ensures existence of all blocks required for correct stencil operations



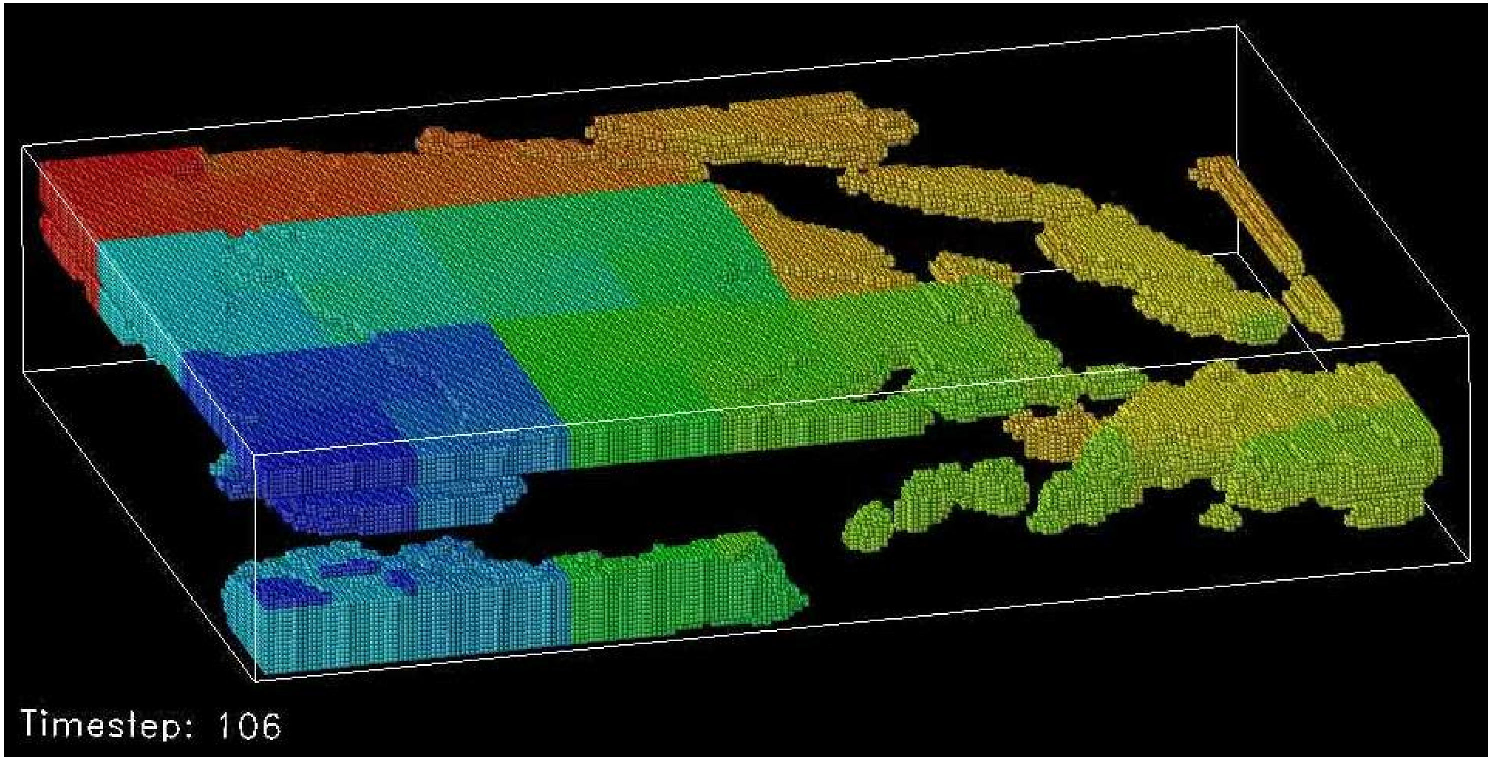
Framework FD4: Dynamic Load Balancing

- When blocks are added or removed (adaptive block mode)
- When load balance decreases below a certain limit
- User can assign each block a weight, e.g. computation time of the block
- Two partitioning methods:
  - Hilbert space-filling curve (SFC) partitioning [Sagan94]
  - Graph partitioning using ParMETIS
- SFC preferred because graph partitioning has much higher overhead



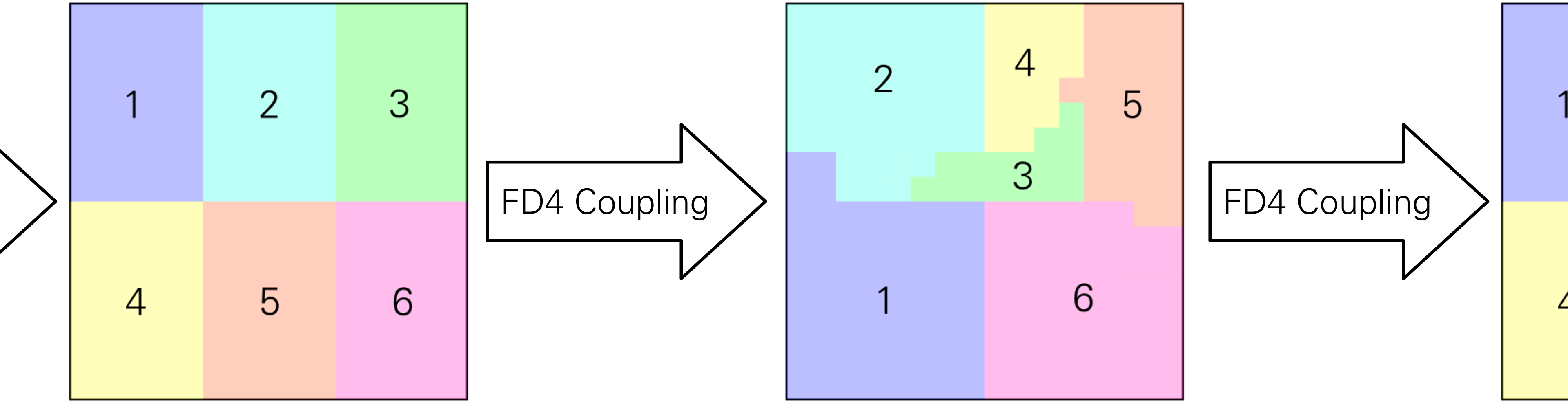


Framework FD4: Dynamic Load Balancing Movie



- Overhead test of adaptive block mode and load balancing [Lieber10]
- FD4 adapts to cloud formation in COSMO weather model
- Real-life scenario, 249 x 174 x 50 grid, 256 processes

Application of FD4: COSMO-SPECS+FD4



COSMO

Computes dynamics

Static MxN partitioning

FD4

Send data to SPECS grid:

u, v, w, T, p, ρ, q_v

SPECS

Computes Microphysics

Data dynamically balanced by FD4

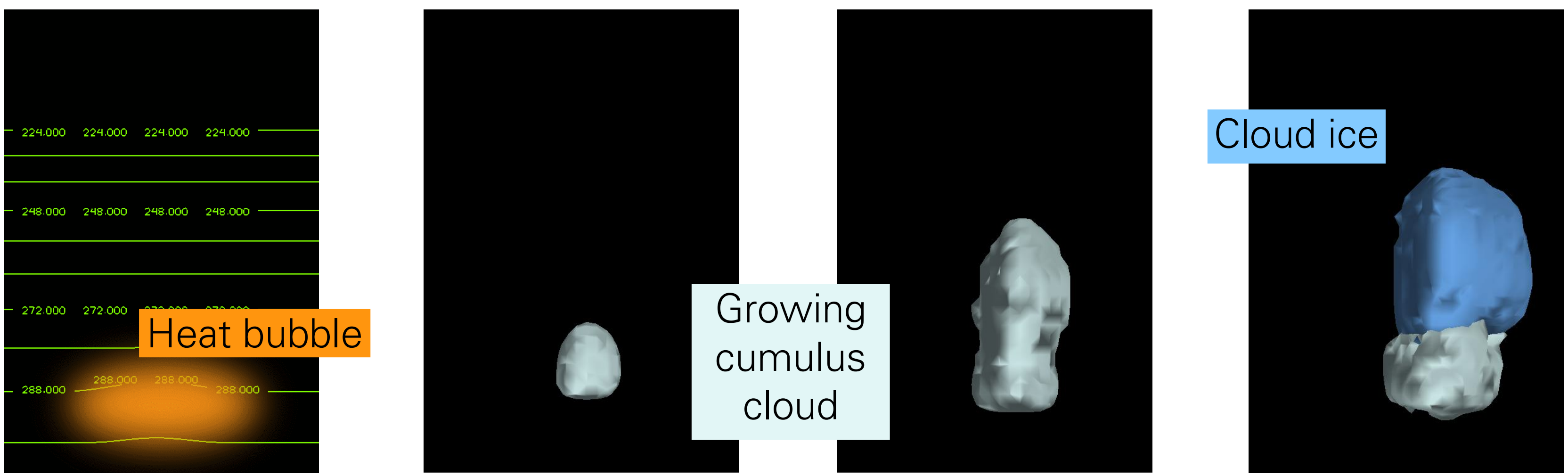
FD4

Receive data from SPECS grid:

ΔT, q_v, q_c, q_i

Application of FD4: Benchmark Case

- Comparing original COSMO-SPECS with COSMO-SPECS+FD4
- Test scenario: heat bubble results in growth of cumulus cloud
- 30 min forecast time
- Vertical grid: 48 nonuniform height levels (up to 18 km)
- Horizontal grid: 32 x 32, 1km resolution
- 2 x 2 x 4 = 16 grid cells per FD4 block

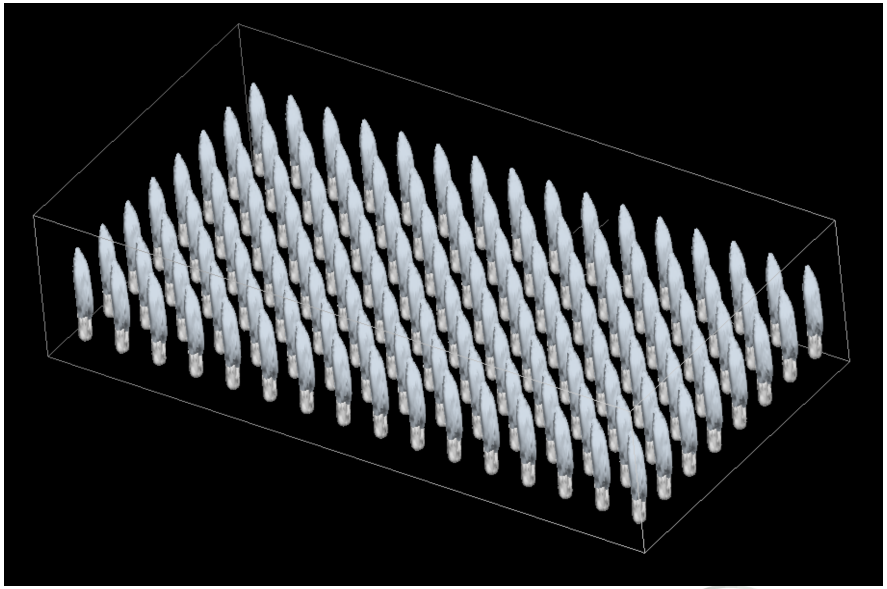


t = 0 min | t = 10 min | t = 20 min | t = 30 min

Application of FD4: Weak Scaling Benchmark Setup

- Weak scaling: problem size per process = constant
- *Replication scaling* benchmark method
  - Create identical copies of same physical problem (i.e. cloud) when scaling up the grid size
- Each 32 x 32 horizontal grid tile initialized with a heat bubble

# Proc.	Grid size	# Replicated clouds	# FD4 blocks
256	32x32	1x1	3072
512	64x32	2x1	6144
1024	64x64	2x2	12 288
...			
32768	512x256	16x8	393 216
65 536	512x512	16x16	786 432



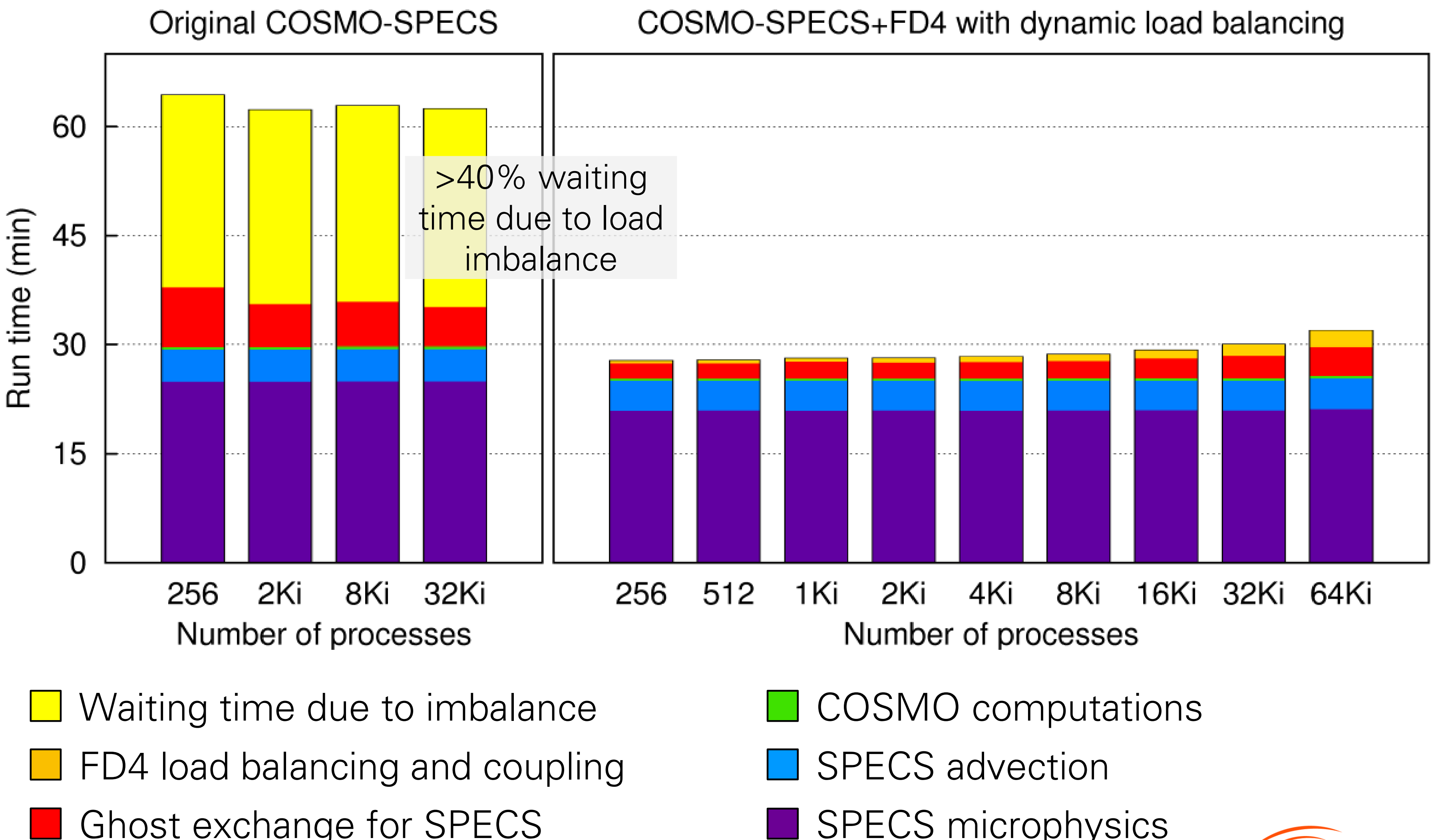
16x8 clouds after 30 min simulation

Application of FD4: Benchmark System IBM BlueGene/P

- IBM BlueGene/P System at Jülich Supercomputing Centre
- 294 912 IBM PowerPC 450 processor cores at 850MHz
- Highly scalable node interconnect
- #5 in the June 2010 Top500 list

JÜLICH
FORSCHUNGSZENTRUM

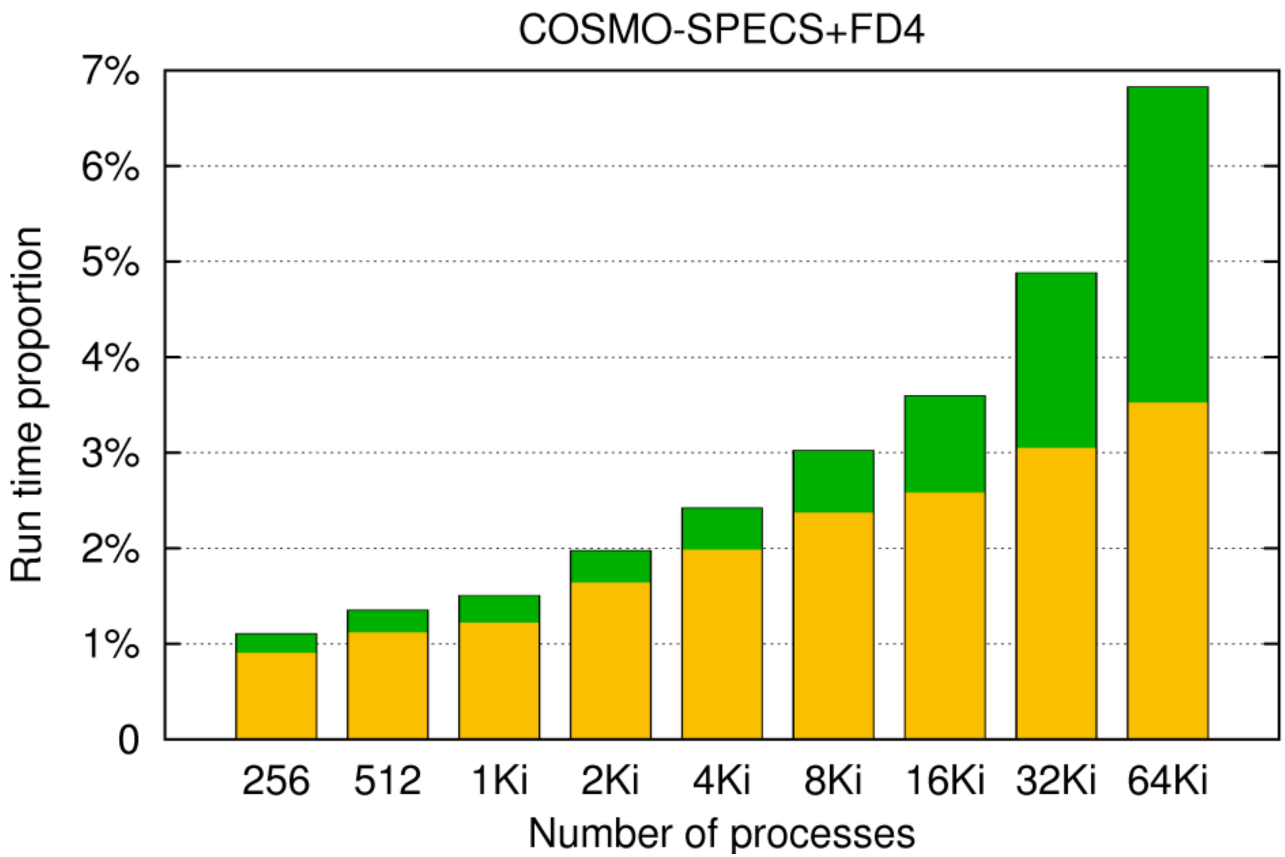
Application of FD4: Performance Comparison



TECHNISCHE UNIVERSITÄT DRESDEN

ZIH Center for Information Services & High Performance Computing

Application of FD4: FD4 Overhead



Coupling (overlap calculation, data transfer)

Dynamic load balancing (partitioning calculation, block migration)

Coupling:
20-fold increase

Dynamic load balancing:
4.5-fold increase

No. of processes and no. of blocks:
256-fold increase

Conclusion & Outlook

- FD4 provides highly scalable dynamic load balancing and coupling for multiphase models
- Scalability to 10 000s of processes
- COSMO-SPECS performance increased significantly by FD4
- FD4 not limited to meteorology
- Freely available at http://www.tu-dresden.de/zih/clouds

- Next steps:
  - Multirate time stepping in COSMO-SPECS+FD4
  - Apply adaptive block mode in COSMO-SPECS+FD4
  - Parallel I/O in FD4

Thank you for your attention!

Acknowledgments

- COSMO Model: German Weather Service (DWD)
- Access to IBM BlueGene/P: Jülich Supercomputing Centre (JSC)
- Funding: German Research Foundation (DFG)

[Grützun08] V. Grützun, O. Knoth, and M. Simmel. *Simulation of the influence of aerosol particle characteristics on clouds and precipitation with LM–SPECS: Model description and first results.* Atmos. Res., 90:233–242, 2008.

[Lieber08] M. Lieber and R. Wolke. *Optimizing the coupling in parallel air quality model systems*, Environ. Modell. Softw., 23:235-243, 2008

[Lieber10] M. Lieber, R. Wolke, V. Grützun, M.S. Müller, and W.E. Nagel. *A framework for detailed multiphase cloud modeling on HPC systems*, in Parallel Computing Vol. 19, 281-288, IOS Press, 2010.

[Pinar04] A. Pinar and C. Aykanat. *Fast optimal load balancing algorithms for 1D partitioning.* J. Parallel Distrib. Comput., 64(8):974-996, 2004.

[Sagan94] H. Sagan. *Space-filling curves*, Springer, 1994

[Simmel06] M. Simmel and S. Wurzler. *Condensation and activation in sectional cloud microphysical models*, Atmos. Res., 80:218-236, 2006.

[Teresco06] J.D. Teresco, K.D. Devine, and J.E. Flaherty. *Partitioning and Dynamic Load Balancing for the Numerical Solution of Partial Differential Equations,* in Numerical Solution of Partial Differential Equations on Parallel Computers, pages 55-88, Springer, 2006.