A framework for detailed multiphase cloud modeling on HPC systems

Matthias LIEBER [a,1], Ralf WOLKE [b], Verena GRÜTZUN [b], Matthias S. MÜLLER [a], and Wolfgang E. NAGEL [a]

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

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

Abstract. Cloud processes are appreciated to be of increasing importance to the comprehension of the atmosphere. Therefore, weather models have recently been extended by detailed spectral descriptions of cloud processes. However, the high computational costs hinder their practical application. This paper introduces the novel framework FD4 (Four-Dimensional Distributed Dynamic Data structures), which is developed to parallelize and couple cloud models to atmospheric models in an efficient way and to enable a higher scalability on HPC systems. Results of first tests with the regional forecast model COSMO are presented.

Keywords. multiphase cloud modeling, spectral microphysics, high performance computing, parallelization framework, load balancing, model coupling

Introduction

Clouds are of significant importance to the atmosphere due to their influence on the radiation budget, the hydrological cycle, scavenging, and wet deposition processes, as well as aqueous-phase chemical reactions [9,24]. Nevertheless, cloud processes represent one of the major uncertainties in current weather forecast, air quality, and climate models [18]. Most of today's weather models incorporate cloud microphysical processes based on the *bulk* approach. Each of the hydrometeor classes (e.g. water droplets, graupel, and snow) is represented by its bulk mass only, while neglecting the size distribution of the particles. However, several studies emphasize the importance of a size-resolving approach [5,15]. Such *spectral* microphysical models explicitly characterize the size distribution of the hydrometeors by applying a bin discretization. These models have been applied for process studies only, but not for operational applications because of very high computational costs. However, the next generation of atmospheric model systems has to incorporate more detailed descriptions of cloud processes in order to achieve more realistic forecasts, for example in air quality modeling [8]. This is only possible by taking the space-time heterogeneity of cloud processes into account in order to focus computation time and

[1] Corresponding Author: Matthias Lieber, Center for Information Services and High Performance Computing (ZIH), TU Dresden, 01062 Dresden, Germany; E-mail: matthias.lieber@tu-dresden.de

This work has been funded by the German Research Foundation (DFG), grant No. NA 711/2-1.

memory on interesting (cloudy) regions. To our knowledge, such dynamic techniques have not yet been used for spectral cloud models.

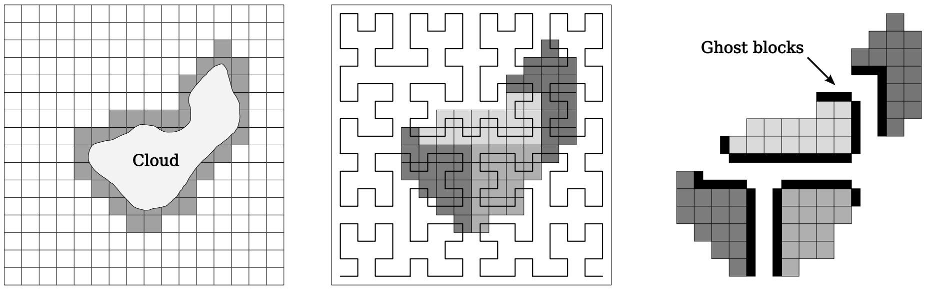
This paper presents first results of our efforts to overcome this deficiency and enable the practical application of such complex model systems. We are developing the software framework FD4 (Four-Dimensional Distributed Dynamic Data structures), which provides an efficient coupling of detailed cloud models to existing atmospheric models. It matches the specific requirements of cloud models in terms of data structures and load balance to achieve better performance on HPC systems than previous approaches. Furthermore, FD4 is designed to be generally applicable to multiphase modeling in the geosciences and chemical engineering [3].

1. Related work

Spectral bin microphysics have been introduced in the 3D atmospheric models MM5 [15], WRF [14], and COSMO [6]. In all three cases, the weather model's framework was used for storage and static domain decomposition of the hydrometeor size distributions. Due to the high computational costs, these detailed cloud models have been used only for process studies until now.

The space-time heterogeneity of cloud processes demands adaptive simulation methods to limit the computational costs. Today, space-time heterogeneity is handled with adaptive mesh refinement (AMR). Several frameworks for parallel structured AMR [4] have been developed in the last years, e.g., PARAMESH [16], CHOMBO [21], and ALPS [2]. In the structured AMR approach, the grid is decomposed into blocks which are refined hierarchically depending on solution features. These blocks are also used for dynamic load balancing [22], which is typically based on space-filling curves (SFC) or graph repartitioning methods. Our framework handles spatial adaptivity not by refinement, but by allocating blocks of a fixed resolution only where clouds exist. However, there are many common problems like dynamic data structures, domain decomposition, load balancing, as well as the need for a flexible user interface.

Coupling models of different disciplines is an important approach to more detailed simulations, especially in the Earth Sciences. Therefore, different coupling frameworks for parallel models have been developed. OASIS [23] and PALM [1] focus on coupling independent models whose grids may have a different structure and decomposition. They use a coupler process as intermediate instance for data transformation with a potential trade-off on performance, but benefits in flexibility when assembling a complex system of many models. A direct and thus tighter coupling is provided by MCT [13]. MESSy [10] is a special infrastructure to couple various fine-grained submodels to a general circulation model. Like MESSy, FD4 is focused on a special application domain which allows us to realize specific optimizations of data structures and parallel algorithms. In contrast to OASIS and PALM, our approach favors performance even if this compromises flexibility. This is motivated in the following section.



1.1 Grid adaption to clouds: Only those blocks are allocated which are required to capture the clouds.

1.2 Hilbert SFC partitioning: Allocated blocks are ordered by the SFC and divided into partitions for parallelization.

1.3 Ghost blocks: Ghost cells are allocated at the partition boundaries only, not at every block, to reduce memory consumption.

Figure 1. Illustration of FD4 concepts for a 2D grid with an exemplary cloud.

2. The multiphase cloud modeling framework FD4

Our experiences with the detailed cloud modeling system COSMO–SPECS [6] influenced the design process of the new framework FD4. From the computational point of view, COSMO–SPECS is a straightforward implementation: the microphysics parameterization of the mesoscale forecast model COSMO [19] has been replaced by the spectral cloud model SPECS [20]. The microphysics computation and related boundary communication take up more than 90% of the total runtime of COSMO-SPECS (measured with 100 processes on an SGI Altix 4700). The high computational costs of the spectral bin microphysics model are further degraded by poorly balanced workload. We observed that grid cells within clouds consume more than six times of the calculation time than grid cells in cloudless areas.

Future applications of detailed cloud models demand a high scalability which is only possible by introducing a dynamic load balancing of the microphysics computations. This requires the separation of the cloud data structures from the usually static domain decomposition of the basic meteorological model. Our new framework FD4 handles this separation along with further optimizations which utilize specific properties of spectral cloud models: The data structures are optimized for huge amounts of data per grid cell and the microphysics variables are only allocated where clouds exist to save memory. This unique combination of features will help making detailed cloud modeling more applicable for practical cases.

The framework FD4 is implemented in Fortran 95 and uses MPI for parallelization, which allows a smooth integration in present weather codes. In its current state, the following basic services for multiphase cloud modeling are provided:

- Block-based decomposition and parallelization of a rectangular numerical grid
- Dynamical adaption of the grid to spatial cloud structure
- Dynamic load balancing
- Coupling interface to embed cloud models into meteorological models
- NetCDF4 and Vis5D output

Important features are described in the following subsections.

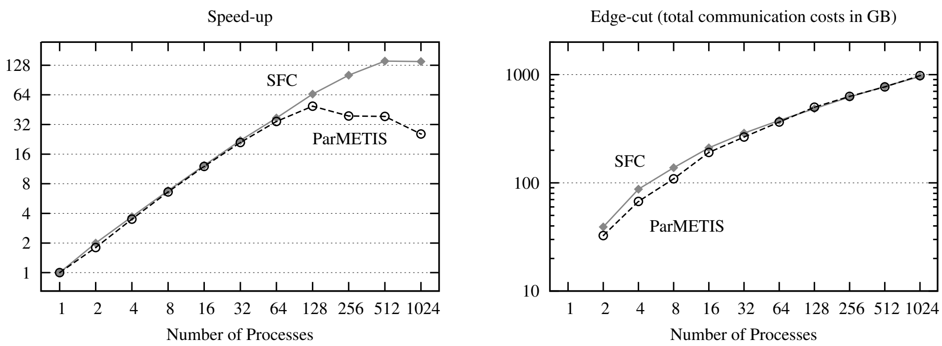


Figure 2. Performance comparison of SFC partitioning and ParMETIS on an SGI Altix 4700 system. The Edge-cut is shown as the total communication volume of the boundary data exchange.

2.1. Adaption to spatial cloud structure

The spectral approach as applied in COSMO-SPECS requires in the order of 1000 variables per grid cell, which potentially results in substantial quantities of wasted memory for cloudless grid cells. FD4 provides a dynamic adaption of the grid to the cloud structure, which means that only the parts of the grid are present where clouds exist and *empty* blocks claim no memory, see Figure 1.1. *Empty* can be defined via a threshold for selected variables, such as, e.g., cloud water concentration. The framework ensures that appropriate data are provided for the numerical stencil around non-*empty* cells. The removal of blocks may cause mass loss, which can be avoided by setting the threshold to zero. However, due to the numerical diffusion of advection schemes, this can lead to a large number of allocated blocks, which limits the advantage of the approach. Therefore, less diffusive advection schemes or volume-of-fluid methods [7] should be applied. In our application, the microphysics reduces the effect of the cloud diffusion by the evaporation of hydrometeors to water vapor.

2.2. Dynamic load balancing

By separating data structures of the cloud model and the atmospheric model, it is possible to apply an individual decomposition with dynamic load balancing for the cloud model. Repartitioning is triggered either when the load balance decreases below a certain limit or when the block structure needs to readapt to the cloud structure, i.e. blocks are added or removed. Two methods of partitioning have been implemented in FD4: partitioning based on the Hilbert space-filling curve (see Figure 1.2) and graph repartitioning using the library ParMETIS [11]. To compare the performance of both approaches, a benchmark simulating the transport of an 'abstract cloud' using a 2nd order advection scheme has been implemented. 1000 variables per grid cell are transported to replicate the memory demands of a detailed spectral model. The parallel block structure dynamically adapts to the cloud structure. Consequently, permanent rebalancing of the blocks is required. Note, that this benchmark is relatively communication-bound, as the advection calculation is computationally inexpensive. The results presented in Figure 2 show that both methods provide comparable partition quality. However, the SFC-based load balancing scales much better. The reason is the faster calculation of the new partitioning in comparison to ParMETIS. A detailed analysis using the Vampir tool-set [12] revealed that ParMETIS spends most of the time performing global MPI communication, which becomes more expensive at higher processor numbers.

2.3. Ghost block implementation

Since we are confronted with hundreds of variables per grid cell, only small blocks (in terms of grid cells) will not exceed the processor cache. Additionally, only such small blocks can adapt efficiently to the spatial cloud structure. With a large number of small blocks it is also possible to achieve a more fine-grained load balancing. For these reasons we optimized the framework for blocks containing only a few grid cells but a large amount of variables. First of all, this means that the typical way to allocate blocks with additional so-called ghost cells to store data from neighbor blocks is too costly in terms of memory requirements. For example, in a three-dimensional decomposition with blocks of 4^3 grid cells, $87.5\,\%$ of the storage is consumed for ghost cells when applying two rows of them. Instead of allocating all blocks with ghost cells, FD4 provides only additional ghost blocks at the processor boundaries as shown in Figure 1.3. To access local blocks and ghost cells in the same manner as in the usual approach, the data are copied to a work array before performing stencil computations. This approach of reducing memory requirements has also been implemented as an option in the PARAMESH framework [16].

2.4. Coupling to meteorological models

FD4 features a coupling interface to send variables from a parallel meteorological model to the data structures of the framework and vice versa. As both have different domain decompositions, the geometrically overlapping areas between the partitions of all senders and the partitions of all receivers are determined by the library. The data of these intersections are then exchanged directly between the meteorological model's data fields and the FD4 data structures.

3. Test applications

We have developed exemplary test applications of the framework in order to demonstrate its applicability and analyze its performance. For this purpose, FD4 has been coupled to the weather forecast models COSMO [19] and WRF [17].

3.1. Adaption to the cloud structure of a real-life scenario in COSMO

This test is based on a COSMO real-life scenario covering Saxony, Germany. The computational grid has a horizontal resolution of 1 km and consists of 249×174 cells in 50 vertical layers. The cloud describing variables (cloud water and cloud ice) of the COSMO model are transmitted to the FD4 framework using its coupling interface. There, the block structure adapts to the cloud structure and is partitioned as depicted in Figure 3. No computations are carried out on the FD4 data structures. The overhead of the framework for block adaption, partitioning, and coupling data transfer has been measured for a one hour forecast with a time step of 10 s. The framework performs basically three tasks for each of the 360 time steps:

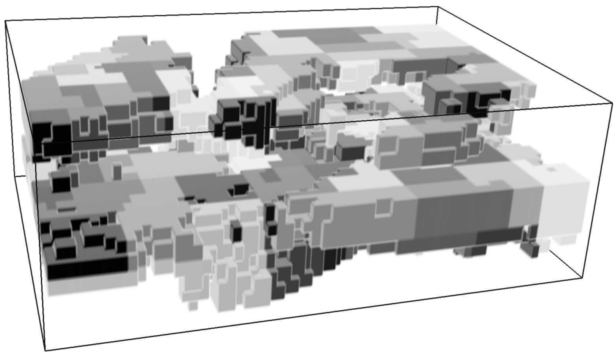


Figure 3. Visualization of the FD4 cloud partitioning for a real-case simulation. The blocks are 4^3 grid cells in size and the partitioning for 64 processes is calculated using the Hilbert space-filling curve.

1. Determine required blocks: Find out which blocks are required to cover all grid cells for which cloud water or cloud ice concentration is not zero in COSMO. Additionally, ensure that a boundary of two grid cells around each of the 'cloudy' cells is present in the block structure.
2. SFC partitioning: Calculate a balanced partitioning of these blocks using the Hilbert SFC.
3. Coupling: Transfer cloud variables from COSMO's partitions to the FD4 block structure.

Note, that the clouds show such high dynamics that the block structure needs to readapt every time step. The benchmark has been run with different block sizes: medium sized blocks with 4^3 grid cells and very small blocks with 2^3 grid cells. In the first case, 11 427 blocks were present in average over the runtime compared to 63 597 blocks in the latter case. Figure 4 shows the results measured on an AMD Opteron cluster with SDR Infiniband interconnect. The determination of required blocks scales only little, as the algorithm's speed depends on the partition size as well as on the total number of blocks. At high processor numbers, collective communication with `MPI_Allgatherv` becomes very costly. The SFC partitioning time depends only on the number of blocks and does not scale. However, in comparison to ParMETIS, the SFC partitioning is much faster. The coupling transfer, which is mainly communication-bound, scales well to 256 processes. Further tuning of the partition matching algorithm is necessary to enable a higher scalability. In summary, the framework's overhead highly depends on the number of blocks, which is a critical issue since we are aiming to apply relatively small blocks.

3.2. Tracer simulation with WRF

As a proof of concept demonstration, FD4 has been coupled to the Weather Research and Forecasting Model (WRF). The WRF model provides an API for I/O and model coupling, which is used to write model output in various formats as well as for coupling of, e.g., ocean models to WRF. Based on this API, we implemented a new coupling package to transfer data fields from WRF to the data structures of FD4. As an exemplary application, the wind fields computed by WRF are used to transport a passive tracer which represents the prospective cloud variables. The tracer is transported with an advection scheme implemented and parallelized with FD4. FD4 adapts the block structure dynamically to

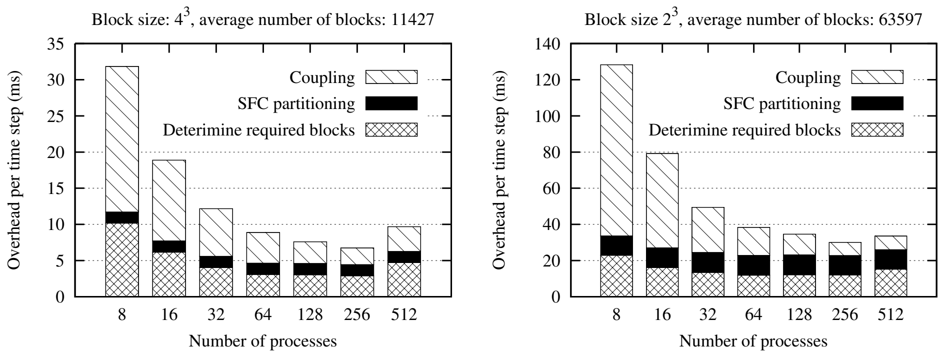


Figure 4. Performance of FD4 in the COSMO test application. The overhead of FD4 components is shown as average time per process and time step for different block sizes.

the propagation of the tracer. This example is already very close to the application of the framework for the coupling of a detailed cloud model to an atmospheric model.

4. Conclusion and outlook

In this paper, we introduced the computational barriers which hinder the operational application of spectral cloud models in three-dimensional atmospheric models. Current approaches use the static framework of their basic weather code to couple the cloud processes and, thus, do not take the heterogeneity of cloud processes into account. The new parallelization and coupling framework FD4 adapts the grid's memory and partitioning dynamically to the cloud structure, to enable a more efficient coupling of the cloud processes than current implementations. As a next step, we will upgrade COSMO–SPECS to use FD4 for the parallelization and coupling of the cloud model. We expect substantial performance gain due to the dynamic load balancing and the more efficient boundary communication provided by FD4. Further, we will add the ability to perform parallel I/O, which is required to efficiently cope with large amounts of data, as in spectral cloud models. Finally, the most challenging task in developing such dynamic frameworks is to keep the balance between adaptivity in the data structures (aiming at saving memory and computations) and the overhead that comes along with this adaptivity.

References

[1] S. Buis, A. Piacentini, and Damien Déclat. PALM: a computational framework for assembling high-performance computing applications. *Concurrency and Computation: Practice and Experience*, 18(2):231–245, 2006.

[2] C. Burstedde, O. Ghattas, M. Gurnis, G. Stadler, E. Tan, T. Tu, L. C. Wilcox, and S. Zhong. Scalable Adaptive Mantle Convection Simulation on Petascale Supercomputers. In *SC 08*, 2008.

[3] A. Caboussat. Numerical Simulation of Two-Phase Free Surface Flows. *Arch. Comput. Meth. Engng.*, 12:165–224, 2005.

[4] L. F. Diachin, R. D. Hornung, P. Plassmann, and A. M. Wissink. Parallel Adaptive Mesh Refinement. In *Parallel processing for scientific computing*. Cambridge University Press, 2006.

[5] K. M. Fahey and S. N. Pandis. Size-resolved aqueous-phase atmospheric chemistry in a three-dimensional chemical transport model. *J. Geophys. Res.*, 108(D22):4690, 2003.

[6] 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.

[7] D. Hinneburg and O. Knoth. Non-dissipative cloud transport in Eulerian grid models by the volume-of-fluid (VOF) method. *Atmos. Environ.*, 39:4321–4330, 2005.

[8] IGAC. *Science Plan and Implementation Strategy*. IGBP Secretariat, 2006. `http://www.igac.noaa.gov`.

[9] IPCC. *Climate Change 2007: The Physical Science Basis. Contribution of Working Group I to the Forth Assessment Report of the Intergovernmental Panel on Climate Change (IPCC).* Cambridge University Press, Cambridge, UK and New York, USA, 2007.

[10] P. Jöckel, H. Tost, A. Pozzer, C. Brühl, J. Buchholz, L. Ganzeveld, P. Hoor, A. Kerkweg, M. G. Lawrence, R. Sander, B. Steil, G. Stiller, M. Tanarhte, D. Taraborrelli, J. van Aardenne, and J. Lelieveld. The atmospheric chemistry general circulation model ECHAM5/MESSy1: consistent simulation of ozone from the surface to the mesosphere. *Atmos. Chem. Phys.*, 6(12):5067–5104, 2006.

[11] G. Karypis, K. Schloegel, and V. Kumar. *ParMETIS: Parallel graph partitioning and sparse matrix ordering library (Version 3.1)*, 2003. University of Minnesota.

[12] A. Knüpfer, H. Brunst, J. Doleschal, M. Jurenz, M. Lieber, H. Mickler, M. S. Müller, and W. E. Nagel. The Vampir Performance Analysis Tool Set. In *Tools for High Performance Computing*, pages 139–155. Springer, 2008.

[13] J. Larson, R. Jacob, and E. Ong. The Model Coupling Toolkit: A New Fortran90 Toolkit for Building Multiphysics Parallel Coupled Models. *Int. J. High Perf. Comput. Appl.*, 19:277–292, 2005.

[14] B. Lynn, A. Khain, D. Rosenfeld, and W. L. Woodley. Effects of aerosols on precipitation from orographic clouds. *J. Geophys. Res.*, 112:D10225, 2007.

[15] B. H. Lynn, A. P. Khain, J. Dudhia, D. Rosenfeld, A. Pokrovsky, and A. Seifert. Spectral (Bin) Microphysics Coupled with a Mesoscale Model (MM5). Part I: Model Description and First Results. *Mon. Weather Rev.*, 133:44–58, 2005.

[16] P. MacNeice, K. M. Olson, C. Mobarry, R. de Fainchtein, and C. Packer. PARAMESH: A parallel adaptive mesh refinement community toolkit. *Comput. Phys. Comm.*, 126:330–354, 2000.

[17] J. Michalakes, J. Dudhia, D. Gill, T. Henderson, J. Klemp, W. Skamarock, and W. Wang. The Weather Research and Forecast Model: Software Architecture and Performance. In *Proceedings of the 11th ECMWF Workshop on the Use of High Performance Computing In Meteorology*, 2004.

[18] P. Monks, P. Borrell, M. Kanakidu, K. Law, M. Millan, and N. Moussiopoulos. Frontiers in the Transport and Transformation of Pollutants. In *The Report of the Barnsdale Expert Meeting 2004*, 2005. `http://www.accent-network.org/portal/project-description/project-archive`.

[19] U. Schättler, G. Doms, , and C. Schraff. *A Description of the Nonhydrostatic Regional COSMO-Model, Part VII: User's Guide*. Deutscher Wetterdienst, 2008. `http://www.cosmo-model.org`.

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

[21] B. Van Straalen, J. Shalf, T. Ligocki, N. Keen, and W. Yang. Scalability challenges for massively parallel AMR applications. In *IPDPS 2009*, 2009.

[22] 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.

[23] S. Valcke and R. Redler. *OASIS4 User Guide (OASIS4_0_2)*. CERFACS, Toulouse, 2006.

[24] V. Wulfmeyer, A. Behrendt, Ch. Kottmeier, U. Corsmeier, et al. The Convective and Orographically-induced Precipitation Study: A Research and Development Project of the World Weather Research Program for Improving Quantitative Precipitation Forecasting in Low-mountain Regions. *Bull. Amer. Meteor. Soc.*, 89(10):1477–1486, 2008.