1 Response to Anonymous referee #1

First of all we wanted to thank you for taking your time to go through the manuscript in detail. Your contribution is very much appreciated. Answers to the specific comments are given below. Appended after the responses is a differential version of the manuscript.

1.1 General remarks

- *This paper describes the work involved in coupling the TenStream solver to a large eddy model. Much of it is concerned with the efficiency of the scheme, which is fine. I do have a problem with the one "scientific" plot though (Fig. 1). The paper states that the evolution of the model is statistically indistinguishable, yet to me Fig. 1 clearly shows significantly more noise in the *domain mean* liquid water path when spectral sampling is turned on. Compare this to Fig. 1 of Pincus and Stevens (2009) which shows that liquid water mixing ratio exhibits about the *same* noise whether or not spectral sampling is used. This would seem to undermine a key result of the paper, and so needs much more investigation. For example: (1) If the LWP noise is detectable in the domain mean, surely the LWP of individual clouds has much greater noise? (2) Can this noise be mitigated, e.g. following my suggestion in item 4 below? In my view this issue needs to be addressed properly for this paper to be fully published.*

  You propose to change the methodology used to sample the spectral range less frequently in time while increasing the spectral sampling. We very much concur with your ideas that the ratio between spatial, temporal and spectral sampling could be improved. Recent works of Pincus and Stevens (2013) and Bozzo et al. (2014) in fact tackle these issues in detail sampling teams of spectral bands at a lower frequency in time. Although a lot could and should be done in that direction, we feel that this is beyond the scope of our manuscript because the only point we wanted to make was that the Monte-Carlo Spectral Integration still works if the same spectral band was sampled for all spatial columns. Concerning computational speed, there wouldn't be a difference in our application between sampling one band every time step or two bands every other time steps. We added a paragraph to the manuscript to further clarify why we feel that uniform MCSI is a viable option. Regarding this topic, please also note the more detailed explanations in the specific comments section.

1.2 Specific Comments:

- *Page 9022 Line 21: "energy-rich" to "energetic"*

Changed

- *Page 9024 Lines 9-10: The references here are only to the LMU group, but other groups are working on the same problem, e.g. Tompkins and Di Giuseppe (JAS 2007) and Hogan and Shonk (JAS 2013) for the shortwave.*

  There is a fundamental difference between low-resolution models that need to be aware of sub-grid cloudiness and others where optical properties are supposed to be fully resolved. The TenStream relates to the latter. We added a sentence mentioning sub-grid-cloudiness aware parameterizations.

- *Page 9025, Equation on line 12: The important factor here is which dimension varies fastest in memory - this depends whether the equation is using the Fortran or C convention - please state.*

  The concern is not the memory layout but rather that for 1D radiative transfer solvers, heating rates are actually computed using only one dimensional vectors. We added two sentences to make that more clear.

- *The discussion in section 2.1 considers the case that radiation is run every timestep but just for one g-point. What if N g-points were computed every N timesteps, where N could be 2, 4, or a larger number? Presumably the cost of reordering arrays would be less since it would not be incurred every timestep, and since the clouds only change by a small amount per timestep, very little additional error would be incurred? The heating rates could be applied evenly to a number of timesteps between calls to the radiation scheme, which might even be an improvement in terms of heating-rate noise, highlighted in my main comment at the start of this review.*

  You address two points here. The first one being the discussion in sec. 2.1 about the loop structures: there is no active reordering going on during runtime. The point we wanted to make is that 3D radiative transfer implies changes in function interfaces and data structures: moving from 1D to 3D. This is a direct result of the horizontal coupling and has to be done irrespective of the spectral sampling method.

  Your second remark targets the question if Monte Carlo Spectral Integration may be improved with a different strategy. As we already stated in the general comments section above, this has been studied by Pincus and Stevens (2013) and Bozzo et al. (2014) and we feel that this is out of scope of this manuscript. However, we repeated the experiments on Mistral, additionally with the original MCSI, and the δ-four stream radiative transfer solver and, for better comparison to the work of Pincus and Stevens (2009), plotted the conditionally sampled avg. liquid water content instead of the liquid water path. It is clear from the theoretical framework they proposed, that the uniform version should introduce more noise in the simulation. However, if that noise is unbiased and does not change the overall evolution of the simulation in the sense that e.g the boundary layer depth evolves unchanged, we argue that the principle ideas of LES as well as MCSI are satisfied. We hope the added discussion in in the MCSI section helps to clarify.

- *Page 9030 Line 10: please state the length of a timestep and the number of g points, to indicate how long it takes for all g points to be computed.*

  The scaling experiments were not done using the MonteCarlo Spectral Integration. We added info about the spectral integration and the average time-step.

- *Figure 2: Explain the black/grey bars that form part of the color bar.*

  The black and gray bars show the alpha channel for the volume renderer. This is used to blend out certain value ranges or make objects semi-/transparent. We added the description to the figure.

- *Figure 4: Explain the legend of the panels with reference to Table 1.*

  Good point, done.

- *Listings 1 and 2: These are meaningless to anyone except a user of this specific model. Please write out what this means in English*

  Added the explanation and moved the listings into the appendix

Many thanks,

Fabian Jakub

2 Response to Anonymous referee #2

First of all we wanted to thank you for taking your time to go through the manuscript in detail. Your contribution is very much appreciated. Answers to the specific comments are given at the second part of this response letter. Appended after the responses is a differential version of the manuscript.

2.1 General remarks

- *This manuscript describes progress to couple an explicit solver for three-dimensional radiative transfer with a large-eddy simulation LES hydrodynamic code. Two numerical choices are explored (the iterative solver and the preconditioner) with emphasis on both strong and weak scaling efficiency. Solving the three-dimensional radiative transfer problem (rather than one-dimensional problem in which every model column is treated independently) is incompatible with one of the algorithmic choices underlying the current treatment of radiation in the LES code, namely the Monte Carlo Spectral Integration algorithm in which the spectral interval for each column is chosen randomly. The authors perform simulations to assess whether a weaker version of the MCSI (where spectral points are held constant across the domain but are chosen randomly in time) is still a viable approach to coupling radiation to LES. This is technical work that will enable some potentially very interesting research on whether three-dimensional radiative transfer effects systematically affect large-eddy simulations.*

  Great, right to the point.

- *It's not clear how useful it is to report this work in isolation. The particular performance results for preconditioners and matrix solvers are specific to the problems (including domain size and the amount of cloudiness) and to the computer systems used for testing, while the results on the weak form of MCSI will no doubt need to be revisited for new problems. Personally, I'd advise students of my own to include this material in the subsequent papers describing results, and if GMD has explicit editorial standards for novelty and relevance the editors may want to look closely at whether this manuscript is adequate.*

  We agree that this work focuses on more technical aspects of the model but we understood that this is well within the scope of GMD, "development and technical papers, describing developments such as new parameterizations or technical aspects of running models". We also think that it is important to show that a parameterization is actually running not only in theory, but also in reality, and in particular that the code scales well in a real-world application on a multi-processor machine. Also, the weak form of the MCSI is a new application and we think that it is valuable to show that it works, even if it hasn't been tested for all possible resolution and cloud scenarios. We showed the parallel scaling behavior on three distinctly different architectures and periphery setups. As for the applicability to a range of different model states, the point of the strong scaling experiments is to examine and understand how a particular solver/preconditioner depends on the scene. The fact that the performance of an iterative solver does depend on the complexity of the simulation makes rigorous testing and documenting ever more interesting and important. We feel confident that the presented edge cases put sensible boundaries on the increase in runtime that is to be expected for atmospheric simulations.

- *With that caveat, the manuscript is generally successful at what it attempts to do. It would be improved most by a little pruning and reorganization aimed at more cleanly separating the two classes of issues (weak MCSI vs. algorithmic choices) and providing a more consistent level of detail to support the argument, keeping in mind that readers will come from both the LES and radiative transfer communities. Some general guidance is provided below. The authors might consider a modified hierarchy for the manuscript to reflect the different concepts being explored. To this reader the top-level ideas/headings might be: Introduction LES and Ten-stream models Weak MCSI Numerical scaling Conclusions The introduction might be similarly reorganized to reflect the separate concepts.*

  We agree concerning the structure of the manuscript and made a few modifications accordingly. In particular, the Monte-Carlo-Spectral-Integration part was given an own section, as suggested.

- *The discussion of the broad motivation for the work that radiation influences cloud development, and that three-dimensional radiative transfer is normally neglected could be expanded by three or five sentences so readers understand why the problem is relevant.*

  We added two sentences to elaborate on first order 3D effects.

- *It will likely to be easier to discuss three-dimensional issues, including the need for efficient algorithms (ten-stream) and implementations (numerical issues to be explored here) before MCSI issues because the motivation for examining weak MCSI comes from wanting to use three-dimensional RT. The general motivation for MCSI (that is, most of the discussion on 9023) should be deferred to the section describing the tests of weak MCSI.*

  We agree that the introduction as to why we care about MCSI is rather broad but we feel that the wide introduction helps readers not accustomed to radiative transfer.

- *Section 2.3, and again on 9033: it should be more clear if the experiments with weak MCSI use one- or three-dimensional radiative transfer calculations. On a related note it would be useful to explain why one set of experiments is used to test weak MCSI and another set used to assess performance of the ten-stream solver. Pincus and Stevens 2009 included an experiment in which the mean radiative driving was suppressed and only the noise remained. It would be useful to repeat these experiments with weak MCSI.*

  Yes, we added a sentence stating that the MCSI simulations were done using 1D solvers. The DYCOMS simulation were used in analogy to the work of Pincus and Stevens (2009), whereas the scaling experiment setups are deliberately kept as simple as possible to investigate how the complexity of cloud dynamics influences the radiative transfer solver computationally (see also next point). To put the impact of the increased noise of the uniform MCSI into perspective, we added a more elaborate discussion on the MCSI (see review response letter #1 and MCSI section).

- *What is the point of the clear-sky experiment? One would think that three-dimensional radiative transfer would be irrelevant in the absence of significant scattering, so its not clear what is being tested or learned with these experiments.*

  The strongly forced warm bubble and the clear-sky experiment are the limits how the cloud field may change between calls to the radiation routines. In a real application, both cases occur and with these two setups we test the assumption that reusing an earlier solution may help convergence. We added a second sentence at the first paragraph of strong scaling section to highlight that.

2.2 Specific comments:

- 9022, line–24: *Surely the idea of radiation coupling to cloud dynamics predates Muller and Bony 2015.*

  Of course, the fact that heating drives convective motion is basic physics. We wanted to highlight relevant work on cloud radiative interaction. We added a sentence putting the work in context.

- 9024, line 9–10: *formatting of references is incorrect*

  Corrected.

- 9024, line 13: *It would be kind to add one sentence explaining how the ten-stream solver works for those not familiar.*

  Added a short description to clarify the key concepts and steps of the TenStream solver.

- 9026, line 21: *This is an abrupt transition. It also sounds a bit like advertising.*

  Rewrote the iterative solver paragraph with a more general introduction.

- 9029, lines 16–26: *The explanation of weak and strong scaling is valuable but could be 50 percent shorter.*

  Thanks, we also feel that a thorough introduction is necessary for the parallelization novice. We trimmed two sentences at the beginning and the end.

- 9031, line 3: *"Retrieving the transport coefficients from the look-up table" ... what transport coefficients? what lookup table? Readers who don't know the ten-stream model well are here left behind.*

  Added a brief explanation what the transport coefficients are and why we need them.

- 9032, line 6: *What is the Mistral computer?*

  Added a reference to the table that lists the respective computing machines.

- 9032, line 15: *Pure speculation about the causes for reduced efficiency is not particularly helpful.*

  In order to really know what causes sub-optimal performance on a machine it is imperative to have a rigorous FLOP and memory model. This however is a huge undertaking for complex algorithms. While we can not separate the individual reasons for inefficiencies we feel that it is nevertheless helpful for the reader to know what the possible mechanisms are.

Many thanks,

Fabian Jakub

References

Bozzo, A., Pincus, R., Sandu, I., and Morcrette, J.-J.: Impact of a spectral sampling technique for radiation on ECMWF weather forecasts, Journal of Advances in Modeling Earth Systems, 6, 1288–1300, doi:10.1002/2014MS000386, URL `http://dx.doi.org/10.1002/2014MS000386`, 2014.

Pincus, R. and Stevens, B.: Monte Carlo spectral integration: A consistent approximation for radiative transfer in large eddy simulations, Journal of Advances in Modeling Earth Systems, 1, doi:10.3894/JAMES.2009.1.1, 2009.

Pincus, R. and Stevens, B.: Paths to accuracy for radiation parameterizations in atmospheric models, Journal of Advances in Modeling Earth Systems, 5, 225–233, doi:10.1002/jame.20027, URL http://dx.doi.org/10.1002/jame.20027, 2013.

Manuscript prepared for Geosci. Model Dev.
with version 2014/09/16 7.15 Copernicus papers of the LaTeX class copernicus.cls.
Date: 7 March 2016

3D Radiative Transfer in Large-Eddy Simulations – Experiences coupling the TenStream solver to the UCLA–LES

Fabian Jakub[1] **and Bernhard Mayer**[1]

[1]LMU Munich , Theresienstr.37 80333 Munich

Correspondence to: Fabian Jakub (fabian.jakub@physik.uni-muenchen.de)

Abstract. The recently developed three dimensional TenStream radiative transfer solver was integrated into the UCLA–LES cloud resolving model. This work documents the overall performance of the TenStream solver as well as the technical challenges migrating from 1D schemes to 3D schemes. In particular the employed Monte-Carlo-Spectral-Integration needed to be re-examined in conjunction with 3D radiative transfer. Despite the fact that the spectral sampling has to be performed uniformly over the whole domain, we find that the Monte-Carlo-Spectral-Integration remains valid. To understand the performance characteristics of the coupled TenStream solver, we conducted weak- as well as strong-scaling experiments. In this context, we investigate two matrix-preconditioner (GAMG and block-jacobi ILU) and find that algebraic multigrid preconditioning performs well for complex scenes and highly parallelized simulations. The TenStream solver is tested for up to 4096 cores and shows a parallel scaling efficiency of 80 % to 90 % on various supercomputers. Compared to the widely employed 1D δ-Eddington two-stream solver, the computational costs for the radiative transfer solver alone increases by a factor of five to ten.

1 Introduction

To improve climate predictions and weather forecasts we need to understand the delicate linkage between clouds and radiation. A trusted tool to further our understanding in atmospheric science is the class of models known as large-eddy-simulations (LES). These models are capable of resolving the most energy-rich energetic eddies and were successfully used to study boundary layer structure as well as shallow and deep convective systems.

Radiative heating and cooling drives convective motion (Muller and Bony, 2015) and influences cloud droplet growth and microphysics (Harrington et al., 2000; Marquis and Harrington, 2005). (Harrington et al., 2000; Marquis and Harrington, 2005). Recent work suggests that cloud radiative feedbacks may also play an important role in atmospheric aggregation (Muller and Bony, 2015). One aspect that has, until now, been studied only briefly is the role of three dimensional radiative transfer. One dimensional radiative transfer by definition ignores effects such as cloud side illumination, displaced cloud shadows and horizontal energy transport in general. While it is clear that the neglect of these three dimensional effects lead to big errors in heating rates, the question if and how much this has an effect on cloud formation is not yet settled (Schumann et al., 2002; Di Giuseppe and Tompkins, 2003; O'Hirok and Gautier, 2005; Frame et al., 2009; Petters, 2009).

While radiative transfer is probably the best understood physical process in atmospheric models it is extraordinarily expensive (computationally) to couple fully three dimensional radiative transfer solvers to LES models.

One reason for the computational complexity involved in radiative transfer calculations is the fact that solvers are not only called once per time step but the radiative transfer has to be integrated over the solar and thermal spectral ranges. A canonical approach for the spectral integration are so called "correlated-k" approximations ((Fu and Liou, 1992; Mlawer et al., 1997)) (Fu and Liou, 1992; Mlawer et al., 1997) where instead of expensive line-by-line calculations, the spectral integration is done with typically one to two hundred spectral bands.

However, even when using simplistic 1D radiative transfer solvers and correlated-k methods for the spectral integration the computation of radiative heating rates is very demanding. As a consequence, radiation is usually not calculated at each time step but rather updated infrequently. This is problematic, in particular in the presence of rapidly changing clouds.

Further strategies are needed to render the radiative transfer calculations computationally feasible.

One such strategy was proposed by Pincus and Stevens (2009) who state that thinning out the calling frequency temporally is equivalent to a sparse sampling of spectral intervals. They proposed not to calculate all spectral bands at each and every time step but rather to pick one spectral band randomly. The error that is introduced by the random sampling is assumed to be statistical and uncorrelated and should not change the overall course of the simulation. Their algorithm is known as Monte-Carlo-Spectral-Integration and is implemented in the UCLA–LES. For each time step and for each vertical column, a spectral band is chosen randomly. This has important consequences for the application of a 3D solver where every column is coupled to its neighbors and it is not meaningful to calculate a different spectral-band in one column and another at the neighboring column. Hence, in order to couple the TenStream solver to the UCLA–LES we need to revisit the Monte-Carlo-Spectral-Integration and check if it is still valid if used with three dimensional solvers.

Another reason for the computational burden is the complexity of the radiation solver alone. Fully three-dimensional solvers such as MonteCarlo (Mayer, 2009) or SHDOM (Evans, 1998) are several orders of magnitude slower than usually employed 1D solvers (e.g. δ-Eddington two-stream~~, Joseph et al. (1976)~~ (Joseph et al., 1976)).

To that end, there is still considerable effort being put into the development of fast parameterizations to account for 3D effects. Recent ~~work includes extensions of 1D solvers to account for~~ works incorporate 3D effects in low resolution sub-grid-cloud aware models (GCM's) by means of overlap assumptions or additional horizontal exchange coefficients (Tompkins and Di Giuseppe, 2007; Hogan and Shonk, 2013). Other parameterizations target high resolution models and propagate radiation on the grid-scale, e.g.~~(Wissmeier et al., 2013)~~ Frame et al. (2009) or Wissmeier et al. (2013) for the solar spectral range or~~(Klinger and Mayer, 2015)~~ Klinger and Mayer (2015) for the thermal.

The TenStream solver (Jakub and Mayer, 2015) is a rigorous, fully coupled, three-dimensional, parallel and, comparably fast radiative transfer approximation. In brief, given the optical properties in a box (absorption and scattering coefficient as well as the asymmetry parameter), the TenStream solver computes the propagation of radiation for each model box using MonteCarlo techniques and stores the respective transport coefficients in a look-up table. The resulting radiative fluxes of one box are then coupled in the vertical (2 streams) as well as in the horizontal directions (8 streams) with their respective neighboring boxes. In this paper we document the steps which were taken to couple the TenStream solver to the UCLA–LES which permits us to drive atmospheric simulations with realistic 3D radiative heating rates.

Section 2 briefly introduces the TenStream solver and the UCLA–LES model. In section 2.2.1 follows a description of two choices of matrix solvers and preconditioners which primarily determine the performance of the TenStream solver.

In section 3 we repeated simulations according to the ~~"~~"Second Dynamics and Chemistry of Marine Stratocumulus field study" (DYCOMS II) to check the validity of Monte-Carlo-Spectral-Integration. Section 4 presents an analysis of the weak- and strong-scaling behavior of the TenStream solver and section 5 discusses the applicability of the model setup for extended cloud-radiation interaction studies.

2 Description of models and core components

2.1 LES model

The LES that we coupled the TenStream solver to is the UCLA–LES model. A description and details of the LES model can be found in Stevens et al. (2005). The model already supports a 1D δ-scaled four-stream solver to compute radiative heating rates. The spectral integration is performed following the correlated-k method of Fu and Liou (1992). We should briefly mention the changes to the model code which were necessary to support a three-dimensional solver. ~~Heating~~

In the case of three dimensional radiative transfer we need to solve the entire domain for one spectral band at once. This is in contrast to one dimensional radiative transfer solvers where the heating rate $H(x,y,\lambda,z)$ is a function of the pixel (x,y), integrated over spectral bands (λ) and solved for one vertical column (z) ~~. In the case of three dimensional radiative transfer we~~ at a time. We therefore need to rearrange the loop structures from

$$H(x,y,\lambda,z) \rightarrow H(\lambda,x,y,z)$$

so that the ~~we may solve the entire domain for one spectral band at once~~ spectral integration over λ is the outermost loop. The fact that we couple the entire domain, and hence need to select the same spectral band for all columns is different from what Pincus and Stevens (2009) did and may weaken the validity of the Monte-Carlo-Spectral-Integration. We will discuss this in section 3. The rearrangement also changes some vectors from 1D to 3D and may thereby introduce copies or caching issues. We find that the change roughly adds a 6 % speed penalty compared to the original single column code (no code optimizations considered). In this paper, calculations are exclusively done using the modified loop structures.

2.2 TenStream RT model

The TenStream radiative transfer model is a parallel approximate solver for the full 3D radiative transfer equation (Jakub and Mayer, 2015). In analogy to a two-stream solver, the

TenStream solver computes the radiative transfer coefficients for up- and downward fluxes and additionally for sideward streams. These transfer coefficients determine the propagation of energy through one box. The coupling of individual boxes is done in a linear equation system which may be written as sparse matrix and is solved using parallel iterative methods. It is difficult to predict the performance of a specific choice of iterative solver or preconditioner beforehand. For that reason, we chose to use the PETSc (Balay et al., 2014) framework which offers a wide range of pluggable iterative solvers and matrix preconditioners. Jakub and Mayer (2015) found that the average increase in runtime compared to 1D two-stream solvers is about a factor of 15. One specifically interesting detail about the use of iterative solvers in the context of fluid dynamics simulations is the fact that we can use the solution at the last time step as an initial guess and thereby speed up the convergence of the solver. Section 4 presents detailed runtime comparisons on various computer architectures and simulation scenarios.

2.2.1 Matrix solver

The resulting equation system of the TenStream solver can be written as a huge but sparse matrix (i.e. most entries are zero). The TenStream matrix is positive definite (strictly diagonal dominant) and asymmetric. Sparse matrices are usually solved using iterative methods because direct methods such as Gaussian-elimination or LU-factorization usually exceed memory limitations. The ~~”~~“Portable, Extensible Toolkit for Scientific Computation” (PETSc Balay et al. (2014)) includes several solvers and preconditioners to choose from.

~~Iterative solvers~~

~~The generalized minimal residual method~~

Iterative solvers

For three dimensional systems of partial differential equations with many degrees of freedom, iterative methods are often more efficient computationally and memory-wise. It is also easier to implement them efficiently on todays compute hardware. The three biggest classes in use today are Conjugate Gradient (CG), Generalized Minimal Residual Method (GMRES) ~~(Saad and Schultz, 1986) is arguably the most versatile iterative method in use today. The reason for its popularity is the robustness and applicability to a wide range of matrices. GMRESworks for symmetric as well a asymmetric matrices, for positive definite and indefinite problems. Another solver suited for asymmetric matrices is~~ and BiConjugate-Gradient methods (Saad, 2003). Given that CG is only suitable for symmetric matrices we will focus on the latter two. In the following we will use the flexible version of GMRES (Saad, 1993) and the “stabilized version of BiConjugate-Gradient-Squared” (Van der Vorst, 1992).

~~Preconditioner~~

Preconditioner

Perhaps even more important than the selection of a suitable solver is the choice of matrix preconditioning. In order to improve the rate of convergence, we try to find a transformation for the matrix that increases the efficiency of the main iterative solver. We can use a preconditioner $\mathcal{P}$ on the initial matrix equation so that it writes:

$$\mathcal{P}\mathcal{A} \cdot x = \mathcal{P}b$$

We can easily see that if $\mathcal{P}$ is close to the inverse of $\mathcal{A}$ the left hand side operator reduces to unity and the effort to solve the system is zero. Of course we cannot cheaply find the inverse of $\mathcal{A}$ but we might find something that resembles $\mathcal{A}^{-1}$ to a certain degree. Obviously for a good cost/efficiency tradeoff the preconditioner should be computationally cheap to apply and considerably reduce the number of iterations the solver needs to converge.

This study suggests two preconditioners for the TenStream solver. We are fully aware that our choices are probably not an optimal solution but they give reasonable results.

The first setup uses a so called stabilized BiConjugate-Gradient solver with incomplete LU factorization (ILU). Direct LU factorizations tend to fill up the sparsity pattern of the matrix and quickly become exceedingly expensive. A workaround is to only fill the preconditioner matrix until a certain threshold of filled entries are reached. A fill level factor of zero prescribes that the preconditioner matrix has the same number of non-zeros as the original matrix. The ILU preconditioner is only available sequentially and in the case of parallelized simulations, each processor applies the preconditioner independently (called ~~”~~“block-jacobi”). Consequently, the preconditioner can not propagate information beyond its local part and we will see in section 4 that this weakens the preconditioner for highly parallel simulations. The PETSc solvers are commonly configured via command-line parameters (see listing 1 for ILU-preconditioning).

The second setup uses a flexible GMRES with geometric algebraic multigrid preconditioning (GAMG). Traditional iterative solvers like Gauss-Seidel or Block-Jacobi are very efficient in reducing the high frequency error. This is why they are called “smoothers”. However, the low frequency errors, i.e. long range errors are dampened only slowly. The general idea of multigrid is to solve the problem on several, coarser grids simultaneously. This way, the smoother is used optimally in the sense that on each grid representation the error which is targeted is rather high frequency error. This coarsening is done until ultimately the problem size is small enough to solve it with direct methods. Considerable effort has been put into the development of black-box multigrid preconditioners. Black-box means in this context that the user, in this case the TenStream solver, does not have to supply the coarse grid representation. Rather, the coarse grids are constructed directly from the matrix representation. The command-line options to use multigrid preconditioning are given in listing 2.

2.3 ~~Monte Carlo Spectral Integration~~

3 Monte Carlo Spectral Integration

There are two reasons why radiative transfer is so expensive computationally. On one hand, a single monochromatic calculation is already quite complex. On the other hand, radiative transfer calculations have to be integrated over a wide spectral range. Even if correlated-k methods are used, the number of radiative transfer calculations is on the order of a hundred. As a result, it becomes unacceptable to perform a full spectral integration at every dynamical time step, even with simple 1D two-stream solvers. This means that in most models, radiative transfer is performed at a lower rate than other physical processes. Pincus and Stevens (2009) proposed that instead of calculating radiative transfer spectrally dense and temporally sparse, one may sample only one spectral band at every model time step. The argument is that the error which is introduced by the coarse spectral sampling is averaged out over time and remains random and uncorrelated. As we mentioned in section 2.1, the three dimensional radiative transfer necessitates to compute the entire domain for one and the same spectral band instead of individual bands for each vertical column. In the following we will refer to the adapted version as uniform Monte-Carlo-Spectral-Integration. It is not clear if the assumptions about the errors being random and uncorrelated still hold true if we reduce the sampling noise. To reason that the Monte-Carlo-Spectral-Integration still holds true in the case of uniform spectral sampling, we repeated the numerical experiment in close resemblance to the original paper of Pincus and Stevens (2009).

There, they used the model setup for the DYCOMS-II simulation (details in Stevens et al. (2005)). They show results for nocturnal simulations. In contrast, here we show results with a constant zenith angle $\theta = 45°$. Radiative transfer is computed with a 1D δ-Eddington two-stream solver. The simulation is started with Monte-Carlo-Spectral-Integration and from 2.5 hours on, also calculated with the full spectral integration ~~. In spite of huge high frequency noise in the heating rates the flow evolves statistically indistinguishable from each other (see fig. 1). This suggests that the~~ and the uniform Monte-Carlo-Spectral-Integration. Note, the good agreement between the full spectral sampling simulation and the one with the original Monte-Carlo-Spectral-Integration ~~is~~ in fig. 1. The uniform formulation of Monte-Carlo-Spectral-Integration leads to high frequency changes in the average liquid water content (LWC). These fluctuation in LWC do however not lead to major differences in the evolution of the boundary layer clouds or turbulent kinetic energy. To put the changes in LWC into perspective,

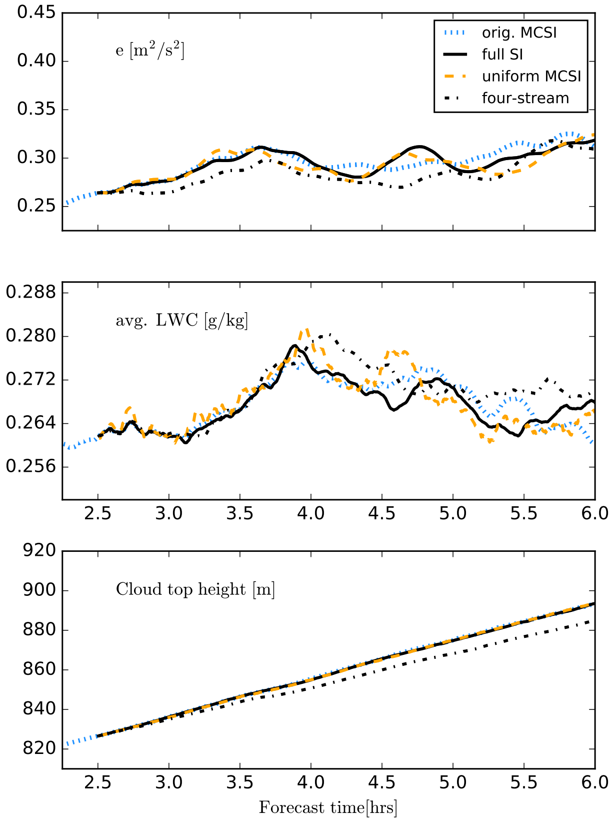


Figure 1: Intercomparison of the DYCOMS II simulation, once forced with the full radiation ~~-~~ (solid line) ~~and -~~, with the original Monte-Carlo-Spectral-Integration ~~-~~ (dotted) and with the uniform version (dashed). The dash-dotted line is a calculation with full spectral integration but with the four-stream solver instead of the two-stream solver. On the top panel, the vertically integrated turbulent kinetic energy, in the middle the ~~vertically integrated~~ mean liquid water ~~path~~ content (conditionally sampled and weighted by physical height) and in the bottom panel the mean cloud top height.

we ran the simulation again with the four-stream solver. While arguably both are good radiative transfer solvers, the choice of the solver leads to bigger and biased changes than the uniform Monte-Carlo-Spectral-Integration. The uniform Monte-Carlo-Spectral-Integration may very well introduce small scale errors but nevertheless seems to be a viable approximation for ~~3D radiative transfer solvers~~this type of simulations. Additionally, we repeated the same kind of experiment for several other scenarios (broken cumulus and deep convection), all confirming the applicability of the uniform Monte-Carlo-Spectral-Integration.

4 Performance Statistics

In the field of High-Performance-Computing it is common to examine the parallel efficiency of an algorithm. To determine the parallel scaling behavior when using an increasing number of processors, one usually conducts two experiments: First, a so called “strong-scaling“” experiment where the problem size stays constant while the number of processors is gradually increased. We speak of linear strong-scaling behavior if the time needed to solve the problem is reduced proportional to the number of used processors. Secondly, a “weak-scaling“” experiment where the problem size and the number of processors are increased linearly. E.g. if we double the domain size, we compute the problem on twice the number of processors whereas the , i.e. the workload per processor is fixed. Linear weak-scaling efficiency implies that the time-to-solution remains constant.

4.1 Strong scaling

We hypothesized earlier (section 2.2) that a good initial guess for the iterative solver results in a faster convergence rate. To test this assumption we performed two strong scaling (problem size stays the same) simulations. One “clear-sky” experiment without clouds in which the difference between radiation calls is minimal and a “warm-bubble” case with a strong cloud deformation and displacement in between time steps. These two situations enclose what the solver may be used for and are hence the extreme cases with respect to the computational effort.

Both scenarios have principally the same setup with a domain length of 10 km at a horizontal resolution of 100 m. The model domain is divided into 50 vertical layers with 70 m resolution at the surface and a vertical grid stretching of 2 %. The atmosphere is moist and neutrally stable (see section 6 for namelist parameters). Simulations are performed with warm cloud microphysicsand , a constant surface temperature, without Monte-Carlo-Spectral-Integration and a dynamic timestep of about 2 s.

Both scenarios are run forward in time for an hour for different solar zenith angles and with varying matrix solvers and preconditioners (presented in section 2.2.1). The difference between the first and the second simulation is the external forcing that was applied. The “clear-sky“”-case is initialized with less moisture, weaker initial wind and no temperature perturbation. No clouds develop in the course of the simulation. In contrast, the second case is initialized with a saturated moisture profile, a strong wind field and a positive, bell shaped, temperature perturbation in the lower atmosphere. The temperature perturbation leads to a rising warm bubble which leads to a cloud shortly after. The initial forcing and latent heat release leads to strong updrafts up to $19\,\mathrm{m\,s^{-1}}$ while the horizontal wind with up to $15\,\mathrm{m\,s^{-1}}$ quickly displaces the cloud sidewards. This strong deformation should give an upper bound on the dissimilarity between calls to the radiation scheme and therefore reduce the quality of the initial guess. To illustrate the general behavior of the strong- and weak scaling experiments, fig. 2 depicts the warm bubble simulation (for the purpose of visualization without initial horizontal wind) – once driven by 1D radiative transfer and once more with the TenStream solver.

Figure 3 presents the increase in runtime of the TenStream solver compared to a 1D calculation. All timings are taken as a best of three and simulations were performed on the IBM Power6 “Blizzard” at DKRZ (Deutsches Klimarechenzentrum), Hamburg in SMT mode[1].

To solve for the direct and diffuse fluxes, the matrix coefficients for the radiation propagation (stored in a 6-dim look-up table) need to be determined for given local optical properties. Retrieving the transport coefficients from the look-up table and the respective linear interpolation (green bar) takes about as long as the 1D radiative transfer calculation alone and is expectedly independent of parallelization and the initial guess of the solution. For larger zenith angles, i.e. lower sun angles, the calculation of direct radiation becomes more and more expensive because of the increasing communication between processors. Note that the computational effort also increases in case of single core runs – the iterative solver needs more iterations because of its treatment of cyclic boundary conditions. The “clear-sky” simulations are computationally cheaper than the more challenging cloud producing “warm-bubble” simulations. In the former, the solver often converges in just one iteration where as in the latter, rather complex case, more iterations are needed. Note that the ILU preconditioning weakens if more processors are used. The ILU is a serial preconditioner and in the case of parallel computations, it is applied to each sub-domain independently. The ILU-preconditioner hence can not propagate information between processors.

The performance of Multi-Grid preconditioning (GAMG) is less affected by parallelization. The number of iterations until converged stays close to constant (independent of the number of processors). The GAMG preconditioning outperforms the ILU preconditioning for many-core systems whereas the setup cost of the coarse grids as well as the interpolation and restriction operators are more expensive if the problem is solved on a few cores only. In summary, we expect the increase in runtime compared to traditionally employed 1D two-stream solvers to be in the range between five to ten times.

4.2 Weak scaling

We examine the weak-scaling behavior using the earlier presented simulation (see section 4.1) but run it only for 10 min. The experiment uses multigrid preconditioning and only performs calculations in the thermal spectral range. The number of grid points is chosen to be 16 by 16 per MPI-rank ($\approx 10^5$ unknown fluxes or $\approx 10^6$ transfer coefficients per processor). The simulations were performed at three different machi-

[1]SMT – Simultaneous Multithreading (2 ranks/core)

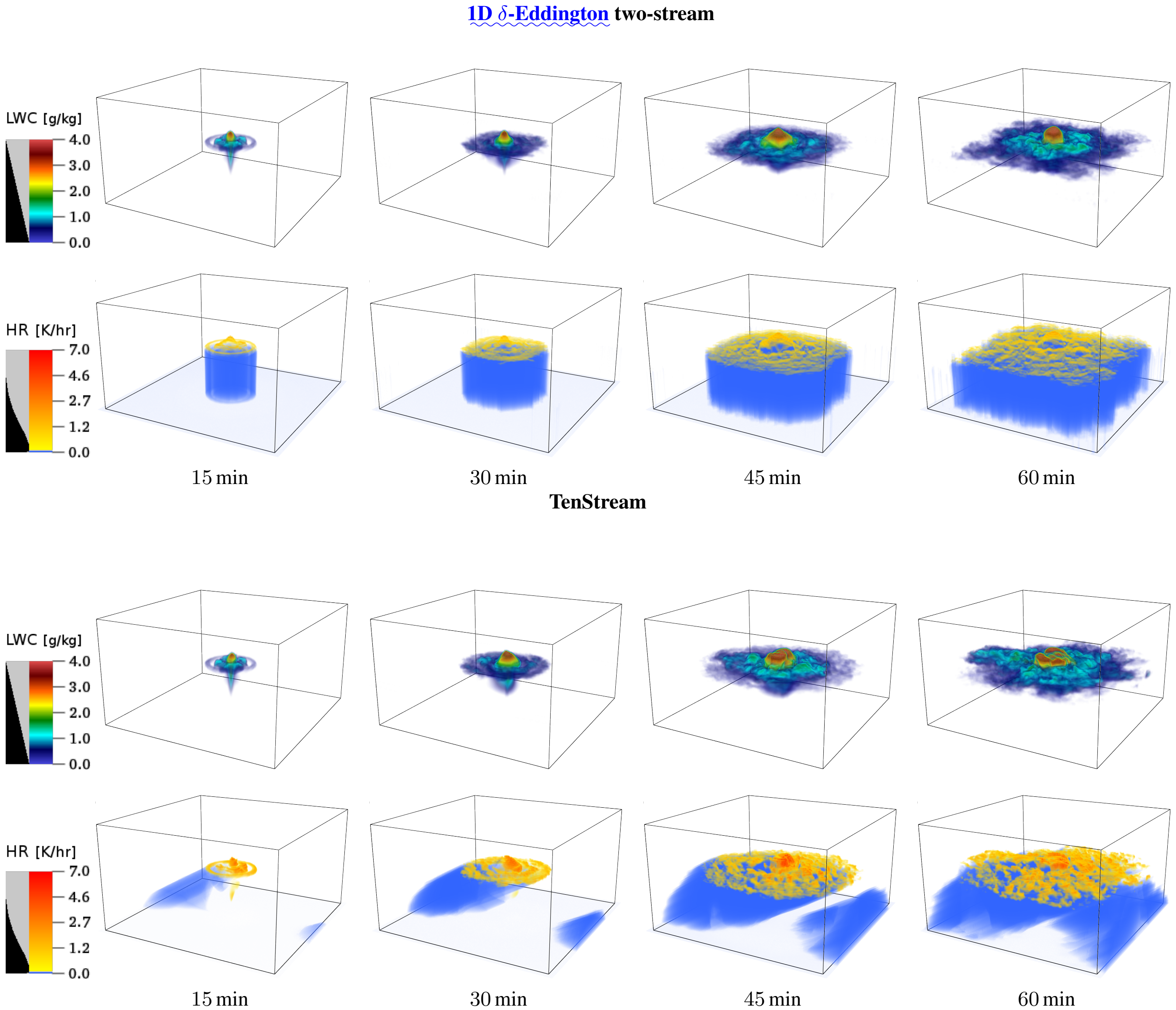


Figure 2: Volume rendered perspective on liquid water content and solar atmospheric heating rates of the warm-bubble experiment (initialized without horizontal wind). The two upper panels depict a simulation which was driven by 1D radiative transfer and the two lower panels show a simulation where radiative transfer is computed with the TenStream solver (solar zenith angle $\theta = 60°$; const. surface fluxes). Three-dimensional effects in atmospheric heating rates introduce anisotropy which in turn has a feedback on cloud evolution. Domain dimensions are $12.8\,\text{km} \times 12.8\,\text{km}$ horizontally and $5\,\text{km}$ vertically at a resolution of $50\,\text{m}$ in each direction. See section 6 for simulation parameters. Gray bar in the legend represents the alpha channel and determines the transparency of the individual colors for the volume renderer.

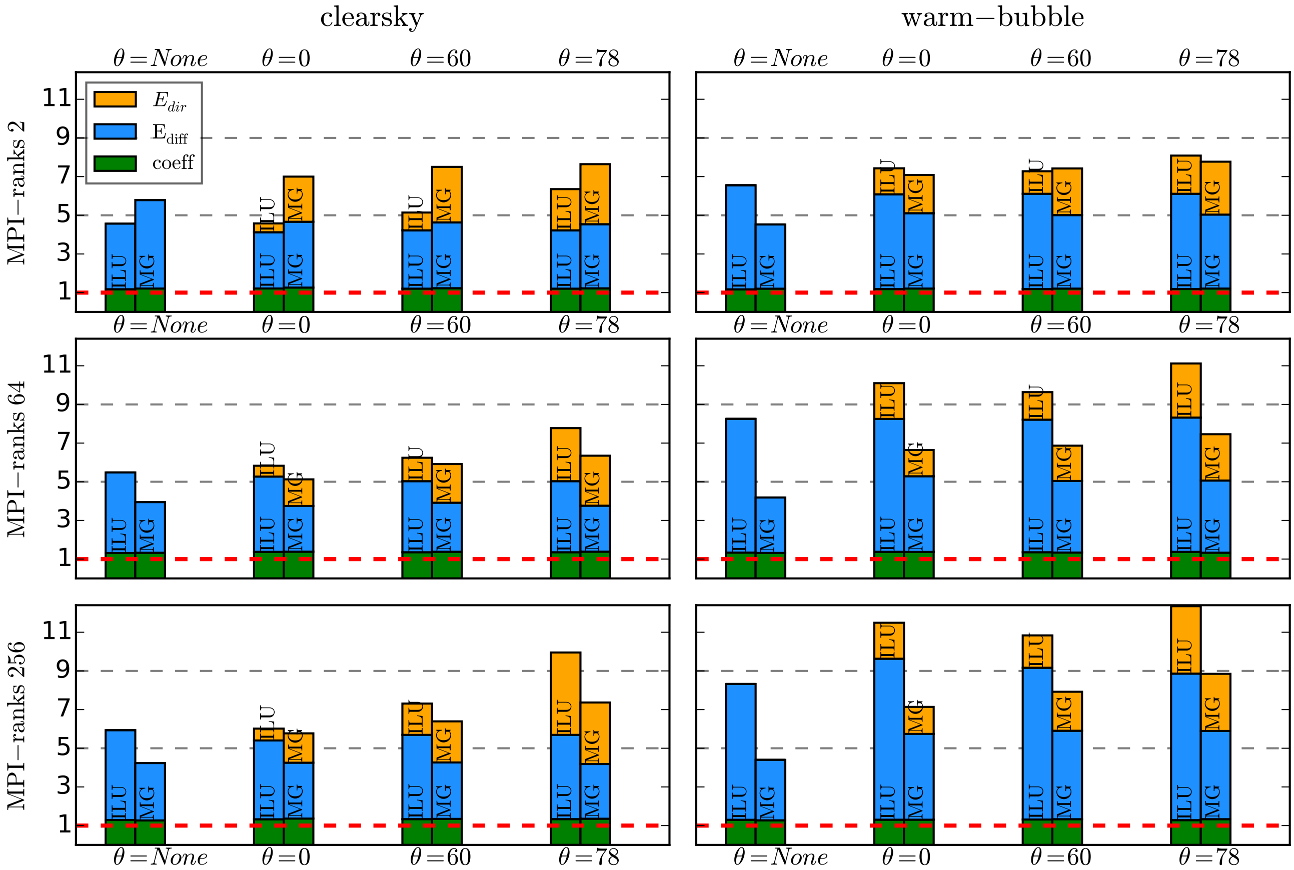


Figure 3: Two strong scaling tests for a clear-sky and a strongly forced scenario. Vertical axis is the increase of computational time normalized to a delta-eddington two-stream calculation (solvers only). Horizontal axis is for different solar zenith angles ($\theta = None$ means thermal only, no solar radiation). The stacked bars denoting time used for the individual components of the solver. "Coeff" meaning the time needed to retrieve and interpolate the transport coefficients. $\mathrm{E_{diff}}$ is the elapsed time that was used to set up the source term and solve for the diffuse radiation; the same for the direct radiation in $\mathrm{E_{dir}}$. The bars are labeled with the corresponding matrix preconditioning.

	Ranks / Node	Cores	Memory-Bandwidth
Mistral	24	2x12@2.5 GHz	112 GB s^{-1}
Blizzard	64	4x 8@4.7 GHz	37 GB s^{-1}
Thunder	16	2x 8@2.6 GHz	76 GB s^{-1}

Table 1: Details on the computers used in this work. Mistral and Blizzard are Intel-Haswell and IBM Power6 supercomputers at DKRZ, Hamburg, respectively. Thunder denotes a Linux Cluster at ZMAW, Hamburg. Columns are the number of MPI ranks used per compute node, the number of sockets and cores, and the maximum memory-bandwidth per node as measured by the streams (McCalpin, 1995) benchmark.

nes/networks (see table 1). Please note that the simulations for ~~the Mistralcomputer~~ Mistral (see table 1) do not fill up the entire nodes (24 cores) since UCLA–LES can currently only run on a number of cores which is a power of two.

Figure 4 presents the weak-scaling efficiency f, defined by:

$$f = \frac{t_{single\,core}}{t_{multi\,core}} \cdot 100\%$$

The scaling behavior can be separated into two regimes: the efficiency on one compute node and the efficiency of the network communication. As long as we stick to one node (fig. 4a), the loss of scaling concerns the 3D TenStream solver as well as the 1D two-stream solver. Reasons for the reduced efficiency may be cache-issues, hyper-threading or memory-bus saturation. The scaling behavior for more than one node (fig. 4b) shows a close to linear scaling for the 1D two-stream solver and a decrease in performance in the case of the TenStream solver. The limiting factor here is network latency and throughput.

5 Conclusions

We described the necessary steps to couple the 3D TenStream radiation solver to the UCLA–LES model. From a technical perspective, this involved the reorganization of the loop structure, i.e. first calculate the optical properties for the entire domain and then solve the radiative transfer.

It was not obvious that the Monte-Carlo-Spectral-Integration would still be valid for 3D radiative transfer. To that end, we conducted numerical experiments (DYCOMS II) in close resemblance to the work of Pincus and Stevens (2009) and find that the Monte-Carlo-Spectral-Integration holds true, even in case of horizontally coupled radiative transfer where the same spectral band is used for the entire domain.

The convergence rate of iterative solvers is highly dependent on the applied matrix-preconditioner. In this work, we tested two different matrix-preconditioners for the TenStream solver: First, an incomplete LU decomposition and secondly the algebraic multigrid-preconditioner, GAMG.We found hat the GAMG preconditioning is superior to the ILU in most cases and especially so for highly parallel simulations.

The increase in runtime is dependent on the complexity of the simulation (how much the atmosphere changes between radiation calls) and the solar zenith angle. We evaluated the performance of the TenStream solver in a weak and strong scaling experiment and presented runtime comparisons to a 1D δ-eddington two-stream solver. The increase in runtime for the radiation calculations ranges from a factor of five up to ten. The total runtime of the LES simulation increased roughly by a factor of two to three. A only twofold increase in runtime allows extensive studies concerning the impact of three dimensional radiative heating on cloud evolution and organization.

This study aimed at documenting the performance and applicability of the TenStream solver in the context of high-resolution modeling. Subsequent work has to quantify the impact of three dimensional radiative heating rates on the dynamics of the model.

6 Code availability

The UCLA–LES model is publicly available at https://github.com/uclales. The calculations were done with the modified radiation interface which is available at git-revision "bbcc4e08ed4cc0789b33e9f2165ac63a7d0573ef".

To obtain a copy of the TenStream code, please contact one of the authors. This study used the TenStream model at git-revision "e0252dd9591579d7bfb8f374ca3b3e6ce9788cd2". For the sake of reproducibility we provide the input parameters for the here mentioned UCLA–LES computations along with the TenStream sources.

Appendix A: Input parameters for the PETSc solvers

Listing 1: BiConjugate-Gradient-Squared iterative solver. The block-jacobi preconditioner does a Incomplete LU preconditioning on each rank with fill level 1 independent of its neighbouring ranks

```
-ksp_type bcgs
-pc_type bjacobi
-sub_pc_type ilu
-sub_pc_factor_levels 1
```

Listing 2: Flexible GMRES solver with algebraic multigrid preconditioning. Use plain aggregation to generate coarse representation (dropping values less than .1 to reduce coarse matrix complexity) and use up to 5 iterations of SOR on coarse grids

```
-ksp_type fgmres
-ksp_reuse_preconditioner
-pc_type gamg
-pc_gamg_type agg
-pc_gamg_agg_nsmooths 0
-pc_gamg_threshold .1
-pc_gamg_square_graph 1
-mg_levels_ksp_type richardson
-mg_levels_pc_type sor
-mg_levels_ksp_max_it 5
```

Acknowledgements. This work was funded by the Federal Ministry of Education and Research (BMBF) through the High Definition Clouds and Precipitation for Climate Prediction (HD(CP)2) project (FKZ: 01LK1208A). Many thanks to Bjorn Stevens and the DKRZ, Hamburg for providing us with the computational resources to conduct our studies.

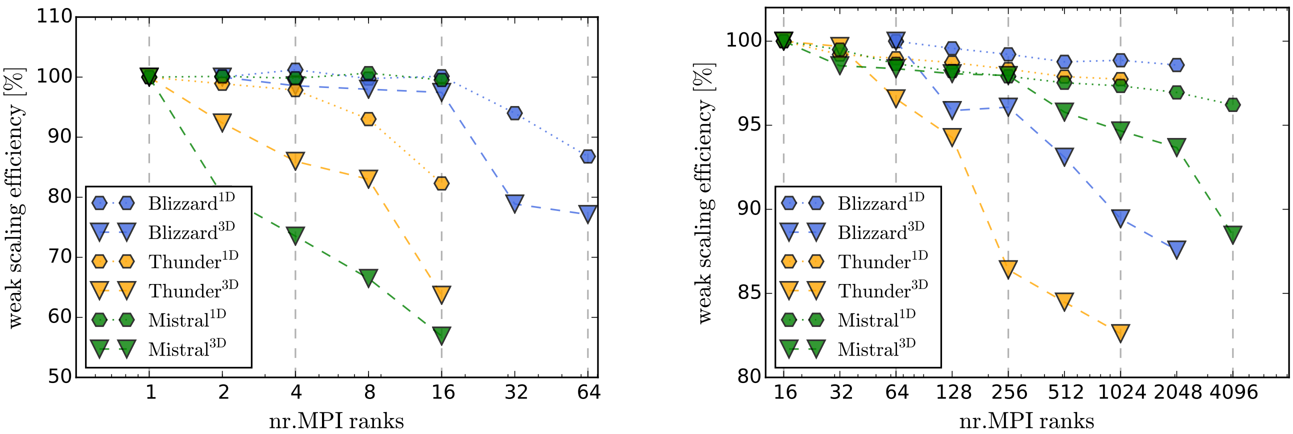
References

Balay, S., Abhyankar, S., Adams, M. F., Brown, J., Brune, P., Buschelman, K., Eijkhout, V., Gropp, W. D., Kaushik, D., Knepley, M. G., McInnes, L. C., Rupp, K., Smith, B. F., and Zhang, H.: PETSc Users Manual, Tech. Rep. ANL-95/11 - Revision 3.5, Argonne National Laboratory, 2014.

Di Giuseppe, F. and Tompkins, A.: Three-dimensional radiative transfer in tropical deep convective clouds, Journal of Geophysical Research: Atmospheres (1984–2012), 108, doi:10.1029/2003JD003392, 2003.

Evans, K. F.: The spherical harmonics discrete ordinate method for three-dimensional atmospheric radiative transfer, Journal of the Atmospheric Sciences, 55, 429–446, doi:10.1175/1520-0469(1998)055<0429:TSHDOM>2.0.CO;2, 1998.

Frame, J. W., Petters, J. L., Markowski, P. M., and Harrington, J. Y.: An application of the tilted independent pixel approximation to



(a) single node

(b) multiple nodes

Figure 4: Weak scaling efficiency running UCLA–LES with interactive radiation schemes. Experiments measure the time for the radiation solvers only (i.e. no dynamics or computation of optical properties). Timings are given as a best of 10 runs. Weak scaling efficiency is given for the TenStream solver (triangle markers) as well as for a two-stream solver (hexagonal markers). (Left) scaling behavior compared to single core computations (remaining on one compute node). (Right) Compute node parallel scaling (normalized against a single node). The individually colored lines correspond to different machines (see table 1 for details) and calculations once done with the δ-eddington two-stream solver (hexagons) and once with the TenStream solver (triangles).

cumulonimbus environments, Atmospheric Research, 91, 127–136, doi:10.1016/j.atmosres.2008.05.005, 2009.

Fu, Q. and Liou, K.: On the correlated k-distribution method for radiative transfer in nonhomogeneous atmospheres, Journal of the Atmospheric Sciences, 49, 2139–2156, doi:10.1175/1520-0469(1992)049<2139:OTCDMF>2.0.CO;2, 1992.

Harrington, J. Y., Feingold, G., and Cotton, W. R.: Radiative impacts on the growth of a population of drops within simulated summertime arctic stratus, Journal of the atmospheric sciences, 57, 766–785, doi:10.1175/1520-0469(2000)057<0766:RIOTGO>2.0.CO;2, 2000.

Hogan, R. J. and Shonk, J. K.: Incorporating the effects of 3D radiative transfer in the presence of clouds into two-stream multilayer radiation schemes, Journal of the Atmospheric Sciences, 70, 708–724, 2013.

Jakub, F. and Mayer, B.: A three-dimensional parallel radiative transfer model for atmospheric heating rates for use in cloud resolving models—The TenStream solver, Journal of Quantitative Spectroscopy and Radiative Transfer, pp. –, doi:http://dx.doi.org/10.1016/j.jqsrt.2015.05.003, http://www.sciencedirect.com/science/article/pii/S0022407315001727, 2015.

Joseph, J., Wiscombe, W., and Weinman, J.: The Delta-Eddington approximation for radiative flux transfer, J. Atmos. Sci., 33, 2452–2459, doi:10.1175/1520-0469(1976)033<2452:TDEAFR>2.0.CO;2, 1976.

Klinger, C. and Mayer, B.: The Neighbouring Column Approximation (NCA)-A fast approach for the calculation of 3D thermal heating rates in cloud resolving models, Journal of Quantitative Spectroscopy and Radiative Transfer, doi:doi:10.1016/j.jqsrt.2015.08.020, 2015.

Marquis, J. and Harrington, J. Y.: Radiative influences on drop and cloud condensation nuclei equilibrium in stratocumulus, Journal of Geophysical Research: Atmospheres (1984–2012), 110, doi:10.1029/2004JD005401, 2005.

Mayer, B.: Radiative transfer in the cloudy atmosphere, in: EPJ Web of Conferences, vol. 1, pp. 75–99, EDP Sciences, doi:10.1140/epjconf/e2009-00912-1, 2009.

McCalpin, J. D.: Memory Bandwidth and Machine Balance in Current High Performance Computers, IEEE Computer Society Technical Committee on Computer Architecture (TCCA) Newsletter, pp. 19–25, 1995.

Mlawer, E. J., Taubman, S. J., Brown, P. D., Iacono, M. J., and Clough, S. A.: Radiative transfer for inhomogeneous atmospheres: RRTM, a validated correlated-k model for the longwave, Journal of Geophysical Research: Atmospheres (1984–2012), 102, 16 663–16 682, doi:10.1029/97JD00237, 1997.

Muller, C. and Bony, S.: What favors convective aggregation and why?, Geophysical Research Letters, 42, 5626–5634, doi:10.1002/2015GL064260, 2015.

O'Hirok, W. and Gautier, C.: The impact of model resolution on differences between independent column approximation and Monte Carlo estimates of shortwave surface irradiance and atmospheric heating rate., Journal of the atmospheric sciences, 62, doi:10.1175/JAS3519.1, 2005.

Petters, J. L.: The impact of radiative heating and cooling on marine stratocumulus dynamics, 2009.

Pincus, R. and Stevens, B.: Monte Carlo spectral integration: A consistent approximation for radiative transfer in large eddy simulations, Journal of Advances in Modeling Earth Systems, 1, doi:10.3894/JAMES.2009.1.1, 2009.

Saad, Y.: A flexible inner-outer preconditioned GMRES algorithm, SIAM Journal on Scientific Computing, 14, 461–469, 1993.

Saad, Y.: Iterative methods for sparse linear systems, Siam, 2003.

Saad, Y. and Schultz, M. H.: GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM Journal on scientific and statistical computing, 7, 856–869, doi:10.1137/0907058, 1986.

Schumann, U., Dörnbrack, A., and Mayer, B.: Cloud-shadow effects on the structure of the convective boundary layer, Meteorologische Zeitschrift, 11, 285–294, 2002.

Stevens, B., Moeng, C.-H., Ackerman, A. S., Bretherton, C. S., Chlond, A., de Roode, S., Edwards, J., Golaz, J.-C., Jiang, H., Khairoutdinov, M., et al.: Evaluation of large-eddy simulations via observations of nocturnal marine stratocumulus, Monthly weather review, 133, 1443–1462, doi:10.1175/MWR2930.1, 2005.

Tompkins, A. M. and Di Giuseppe, F.: Generalizing cloud overlap treatment to include solar zenith angle effects on cloud geometry, Journal of the atmospheric sciences, 64, 2116–2125, 2007.

Van der Vorst, H. A.: Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems, SIAM Journal on scientific and Statistical Computing, 13, 631–644, doi:10.1137/0913035, 1992.

Wissmeier, U., Buras, R., and Mayer, B.: paNTICA: A Fast 3D Radiative Transfer Scheme to Calculate Surface Solar Irradiance for NWP and LES Models., Journal of Applied Meteorology & Climatology, 52, doi:10.1175/JAMC-D-12-0227.1, 2013.