

This discussion paper is/has been under review for the journal Geoscientific Model Development (GMD). Please refer to the corresponding final paper in GMD if available.

3-D radiative transfer in large-eddy simulations – experiences coupling the TenStream solver to the UCLA–LES

F. Jakub and B. Mayer

LMU Munich, Theresienstr. 37, 80333 Munich, Germany

Received: 3 September 2015 – Accepted: 5 October 2015 – Published: 20 October 2015

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

Published by Copernicus Publications on behalf of the European Geosciences Union.

GMDD

8, 9021–9043, 2015

Coupling of the TenStream solver to UCLA–LES

F. Jakub and B. Mayer

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures



Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



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 1-D schemes to 3-D schemes. In particular the employed Monte-Carlo-Spectral-Integration needed to be re-examined in conjunction with 3-D 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 80–90 % on various supercomputers. Compared to the widely employed 1-D 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 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). One aspect that has, until now, been studied only briefly is the

GMDD

8, 9021–9043, 2015

Coupling of the TenStream solver to UCLA–LES

F. Jakub and B. Mayer

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures



Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



role of three-dimensional radiative transfer (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) 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 1-D 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 3-D 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

Coupling of the TenStream solver to UCLA-LES

F. Jakub and B. Mayer

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures



Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



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 1-D solvers (e.g. δ -Eddington two-stream, Joseph et al., 1976).

To that end, there is still considerable effort being put into the development of fast parameterizations to account for 3-D effects. Recent work includes extensions of 1-D solvers to account for 3-D effects, e.g. (Wissmeier et al., 2013) for the solar spectral range or (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 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 3-D radiative heating rates.

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

In Sect. 2.3 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 3 presents an analysis of the weak- and strong-scaling behavior of the TenStream solver and Sect. 4 discusses the applicability of the model setup for extended cloud-radiation interaction studies.

Coupling of the TenStream solver to UCLA-LES

F. Jakub and B. Mayer

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures



Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



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 1-D 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 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 need to rearrange the loop structures from

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

so that we may solve the entire domain for one spectral band at once. 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 Sect. 2.3. The rearrangement also changes some vectors from 1-D to 3-D 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 3-D 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

GMDD

8, 9021–9043, 2015

Coupling of the TenStream solver to UCLA-LES

F. Jakub and B. Mayer

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures



Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



GMDD

8, 9021–9043, 2015

Coupling of the
TenStream solver to
UCLA–LES

F. Jakub and B. Mayer

[Title Page](#)[Abstract](#)[Introduction](#)[Conclusions](#)[References](#)[Tables](#)[Figures](#)[Back](#)[Close](#)[Full Screen / Esc](#)[Printer-friendly Version](#)[Interactive Discussion](#)

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 1-D 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 3 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 (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. GMRES works for symmetric as well as asymmetric matrices, for positive definite and indefinite problems.

Another solver suited for asymmetric matrices is the “stabilized version of BiConjugate-Gradient-Squared” (Van der Vorst, 1992).

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 Sect. 3 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).

Coupling of the TenStream solver to UCLA–LES

F. Jakub and B. Mayer

[Title Page](#)

[Abstract](#)

[Introduction](#)

[Conclusions](#)

[References](#)

[Tables](#)

[Figures](#)



[Back](#)

[Close](#)

[Full Screen / Esc](#)

[Printer-friendly Version](#)

[Interactive Discussion](#)



GMDD

8, 9021–9043, 2015

Coupling of the TenStream solver to UCLA–LES

F. Jakub and B. Mayer

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures



Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



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

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 1-D 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 Sect. 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. It is not clear if the assump-

tions 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 $\theta = 45^\circ$. The simulation is started with Monte-Carlo-Spectral-Integration and from 2.5 h on, also calculated with 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 Monte-Carlo-Spectral-Integration is a viable approximation for 3-D radiative transfer solvers. Additionally, we repeated the same kind of experiment for several other scenarios (broken cumulus and deep convection), all confirming the applicability of the Monte-Carlo-Spectral-Integration.

3 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 workload per processor is fixed. Linear weak-scaling efficiency implies that the time-to-solution remains constant.

Coupling of the TenStream solver to UCLA–LES

F. Jakub and B. Mayer

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures



Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



3.1 Strong scaling

We hypothesized earlier (Sect. 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.

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 Sect. 4 for namelist parameters). Simulations are performed with warm cloud microphysics and a constant surface temperature.

Both scenarios are run forward in time for an hour for different solar zenith angles and with varying matrix solvers and preconditioners (presented in Sect. 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 m s^{-1} while the horizontal wind with up to 15 m s^{-1} quickly displaces the cloud sideways. 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 1-D radiative transfer and once more with the TenStream solver.

Figure 3 presents the increase in runtime of the TenStream solver compared to a 1-D calculation. All timings are taken as a best of three and simulations were performed

GMDD

8, 9021–9043, 2015

Coupling of the TenStream solver to UCLA-LES

F. Jakub and B. Mayer

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures



Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



on the IBM Power6 “Blizzard” at DKRZ (Deutsches Klimarechenzentrum), Hamburg in SMT mode¹.

Retrieving the transport coefficients from the look-up table and the respective interpolation (green bar) takes about as long as the 1-D 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 1-D two-stream solvers to be in the range between five to ten times.

¹SMT – Simultaneous Multithreading (2 ranks/core).

GMDD

8, 9021–9043, 2015

Coupling of the TenStream solver to UCLA–LES

F. Jakub and B. Mayer

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures



Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



3.2 Weak scaling

We examine the weak-scaling behavior using the earlier presented simulation (see Sect. 3.1) 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 machines/networks (see Table 1). Please note that the simulations for the Mistral computer 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:

$$f = \frac{t_{\text{single core}}}{t_{\text{multi core}}} \times 100\%$$

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 3-D TenStream solver as well as the 1-D 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 1-D two-stream solver and a decrease in performance in the case of the TenStream solver. The limiting factor here is network latency and throughput.

4 Conclusions

We described the necessary steps to couple the 3-D 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.

Coupling of the TenStream solver to UCLA–LES

F. Jakub and B. Mayer

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures



Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



Coupling of the TenStream solver to UCLA–LES

F. Jakub and B. Mayer

[Title Page](#)[Abstract](#)[Introduction](#)[Conclusions](#)[References](#)[Tables](#)[Figures](#)[Back](#)[Close](#)[Full Screen / Esc](#)[Printer-friendly Version](#)[Interactive Discussion](#)

It was not obvious that the Monte-Carlo-Spectral-Integration would still be valid for 3-D 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 that 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 1-D 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.

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 repro-

ducibility we provide the input parameters for the here mentioned UCLA–LES computations along with the TenStream sources.

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. 9026
- Di Giuseppe, F. and Tompkins, A.: Three-dimensional radiative transfer in tropical deep convective clouds, *J. Geophys. Res.: Atmos.* (1984–2012), 108, 4741, doi:10.1029/2003JD003392, 2003. 9023
- Evans, K. F.: The spherical harmonics discrete ordinate method for three-dimensional atmospheric radiative transfer, *J. Atmos. Sci.*, 55, 429–446, doi:10.1175/1520-0469(1998)055<0429:TSHDOM>2.0.CO;2, 1998. 9024
- Frame, J. W., Petters, J. L., Markowski, P. M., and Harrington, J. Y.: An application of the tilted independent pixel approximation to cumulonimbus environments, *Atmos. Res.*, 91, 127–136, doi:10.1016/j.atmosres.2008.05.005, 2009. 9023
- Fu, Q. and Liou, K.: On the correlated k-distribution method for radiative transfer in nonhomogeneous atmospheres, *J. Atmos. Sci.*, 49, 2139–2156, doi:10.1175/1520-0469(1992)049<2139:OTCDMF>2.0.CO;2, 1992. 9023, 9025
- Harrington, J. Y., Feingold, G., and Cotton, W. R.: Radiative impacts on the growth of a population of drops within simulated summertime arctic stratus, *J. Atmos. Sci.*, 57, 766–785, doi:10.1175/1520-0469(2000)057<0766:RIOTGO>2.0.CO;2, 2000. 9022
- 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, *J. Quant. Spectrosc. Ra.*, 163, 63–71, doi:10.1016/j.jqsrt.2015.05.003, 2015. 9024, 9025, 9026

Coupling of the TenStream solver to UCLA–LES

F. Jakub and B. Mayer

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures



Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



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. 9024

5 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, *J. Quant. Spectrosc. Ra.*, 168, 17–28, doi:10.1016/j.jqsrt.2015.08.020, 2015. 9024

Marquis, J. and Harrington, J. Y.: Radiative influences on drop and cloud condensation nuclei equilibrium in stratocumulus, *J. Geophys. Res.-Atmos.* (1984–2012), 110, D10205, doi:10.1029/2004JD005401, 2005. 9022

10 Mayer, B.: Radiative transfer in the cloudy atmosphere, *Eur. Phys. J. Conferences* 1, 75–99, doi:10.1140/epjconf/e2009-00912-1, 2009. 9024

McCalpin, J. D.: Memory bandwidth and machine balance in current high performance computers, *IEEE Comp. Soc. Techn. Commit. Comput. Arch. (TCCA) Newslett.*, 19–25, 1995. 9037

15 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, *J. Geophys. Res.-Atmos.* (1984–2012), 102, 16663–16682, doi:10.1029/97JD00237, 1997. 9023

20 Muller, C. and Bony, S.: What favors convective aggregation and why?, *Geophys. Res. Lett.*, 42, 5626–5634, doi:10.1002/2015GL064260, 2015. 9022

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, *J. Atmos. Sci.*, 62, 2939–2951, doi:10.1175/JAS3519.1, 2005. 9023

25 Petters, J. L.: The Impact of Radiative Heating and Cooling on Marine Stratocumulus Dynamics, PSU-eTD, No. 10199, <https://etda.libraries.psu.edu/paper/10199/>, 2009. 9023

Pincus, R. and Stevens, B.: Monte Carlo spectral integration: a consistent approximation for radiative transfer in large eddy simulations, *J. Adv. Model. Earth Syst.*, 5, 2, 225–233, doi:10.3894/JAMES.2009.1.1, 2009. 9023, 9025, 9028, 9029, 9033

30 Saad, Y. and Schultz, M. H.: GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems, *SIAM J. Sci. Stat. Comp.*, 7, 856–869, doi:10.1137/0907058, 1986. 9026

Schumann, U., Dörnbrack, A., and Mayer, B.: Cloud-shadow effects on the structure of the convective boundary layer, *Meteorol. Z.*, 11, 285–294, 2002. 9023

**Coupling of the
TenStream solver to
UCLA–LES**F. Jakub and B. Mayer

[Title Page](#)[Abstract](#)[Introduction](#)[Conclusions](#)[References](#)[Tables](#)[Figures](#)[Back](#)[Close](#)[Full Screen / Esc](#)[Printer-friendly Version](#)[Interactive Discussion](#)

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., Kirkpatrick, P., Lewellen, David C., Lock, Adrian, Müllera, F., Stevens, D. E., Whelan, E., and Zhu, P.: Evaluation of large-eddy simulations via observations of nocturnal marine stratocumulus, *Mon. Weather Rev.*, 133, 1443–1462, doi:10.1175/MWR2930.1, 2005. 9025, 9029

Van der Vorst, H. A.: Bi-CGSTAB: a fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems, *SIAM J. Sci. Stat. Comp.*, 13, 631–644, doi:10.1137/0913035, 1992. 9027

Wissmeier, U., Buras, R., and Mayer, B.: paNTICA: a fast 3D radiative transfer scheme to calculate surface solar irradiance for NWP and LES models, *J. Appl. Meteorol. Climatol.*, 52, 1698–1715, doi:10.1175/JAMC-D-12-0227.1, 2013. 9024

GMDD

8, 9021–9043, 2015

Coupling of the
TenStream solver to
UCLA–LES

F. Jakub and B. Mayer

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.

	Ranks/ Node	Cores	Memory- Bandwidth
Mistral	24	2 × 12@2.5 GHz	112 GBs ⁻¹
Blizzard	64	4 × 8@4.7 GHz	37 GBs ⁻¹
Thunder	16	2 × 8@2.6 GHz	76 GBs ⁻¹

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

◀

▶

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



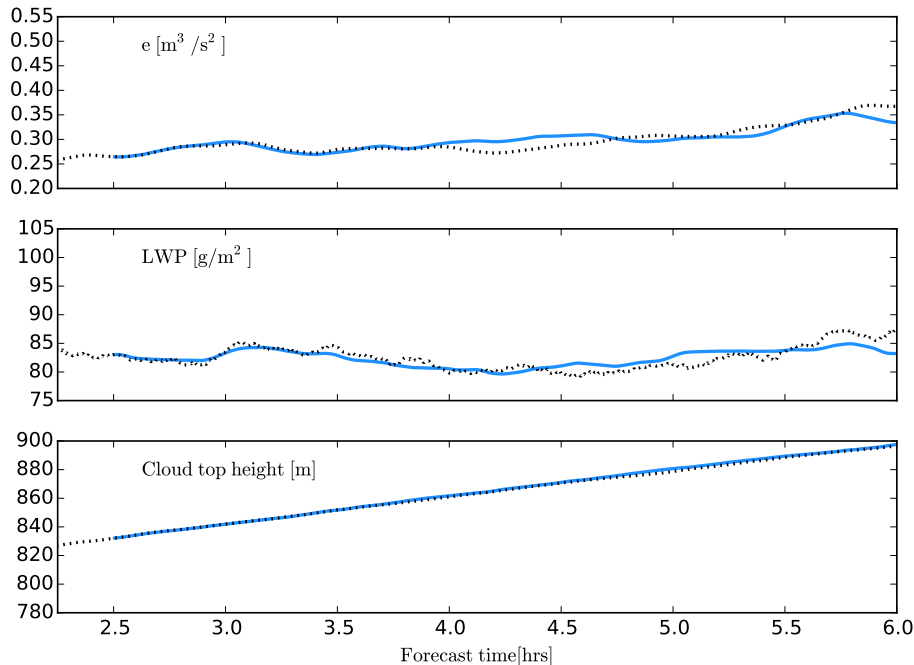


Figure 1. Intercomparison of the DYCOMS II simulation, once forced with the full radiation (solid line) and with the Monte-Carlo-Spectral-Integration (dotted). On the top panel, the vertically integrated turbulent kinetic energy, in the middle the vertically integrated mean liquid water path and in the bottom panel the mean cloud top height.

Coupling of the TenStream solver to UCLA-LES

F. Jakub and B. Mayer

Title Page

Abstract Introduction

Conclusions References

Tables Figures

◀ ▶

◀ ▶

Back Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



Coupling of the TenStream solver to UCLA-LES

F. Jakub and B. Mayer

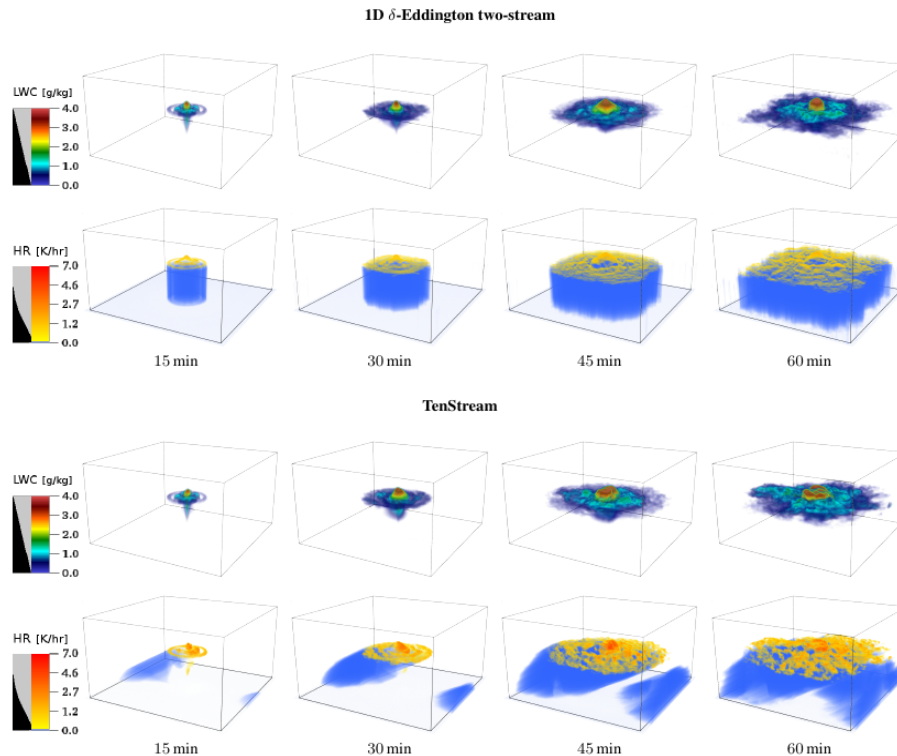


Figure 2. Volume rendered perspective on liquid water content and solar atmospheric heating rates of the warm-bubble experiment (initialized without horizontal wind). The two upper panels depict a simulation which was driven by 1-D radiative transfer and the two lower panels show a simulation where radiative transfer is computed with the TenStream solver (solar zenith angle $\theta = 60^\circ$; 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 km \times 12.8 km horizontally and 5 km vertically at a resolution of 50 m in each direction. See Sect. 4 for simulation parameters.

[Title Page](#)
[Abstract](#)
[Introduction](#)
[Conclusions](#)
[References](#)
[Tables](#)
[Figures](#)
[⏪](#)
[⏩](#)
[◀](#)
[▶](#)
[Back](#)
[Close](#)
[Full Screen / Esc](#)
[Printer-friendly Version](#)
[Interactive Discussion](#)


Coupling of the TenStream solver to UCLA-LES

F. Jakub and B. Mayer

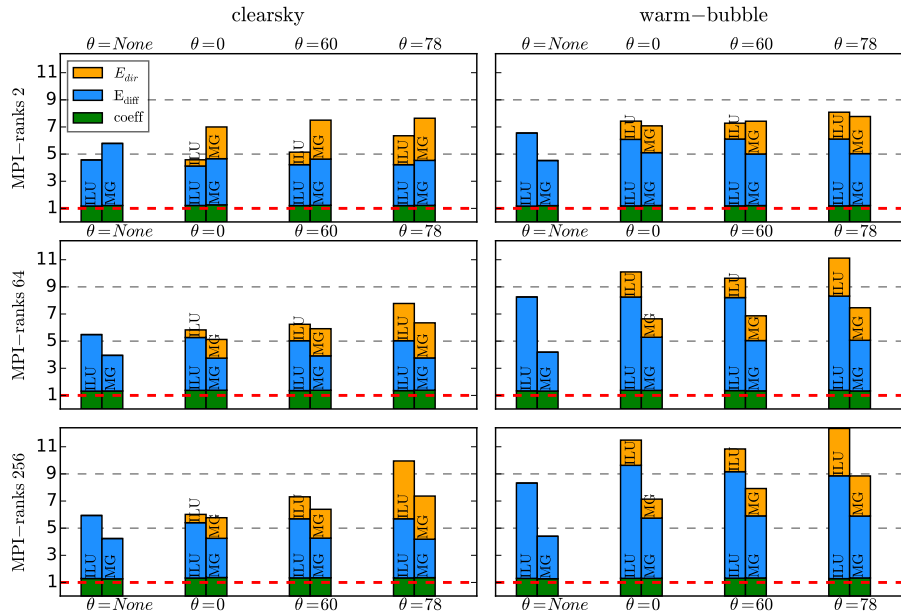


Figure 3. Two strong scaling tests for a clear-sky and a strongly forced scenario. Vertical axis is the increase of computational time normalized to a delta-eddington two-stream calculation (solvers only). Horizontal axis is for different solar zenith angles ($\theta = \text{None}$ means thermal only, 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. 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 E_{dir} . The bars are labeled with the corresponding matrix preconditioning.

Title Page

Abstract Introduction

Conclusions References

Tables Figures

◀ ▶

◀ ▶

Back Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



Coupling of the TenStream solver to UCLA–LES

F. Jakub and B. Mayer

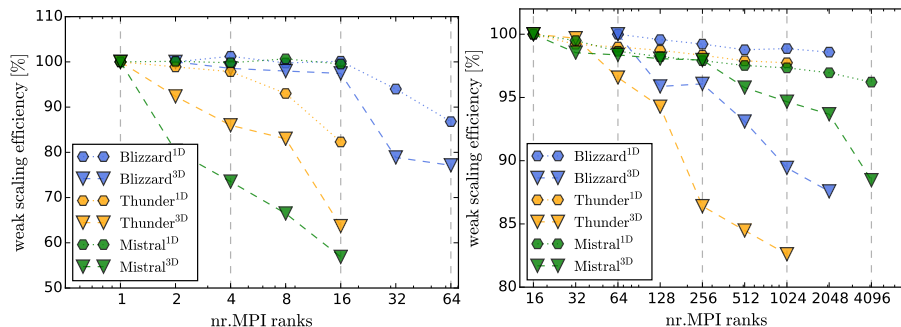


Figure 4. Weak scaling efficiency running UCLA–LES with interactive radiation schemes. Experiments measure the time for the radiation solvers only (i.e. no dynamics or computation of optical properties). Timings are given as a best of 10 runs. Weak scaling efficiency is given for the TenStream solver (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).

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures



Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



GMDD

8, 9021–9043, 2015

Coupling of the TenStream solver to UCLA–LES

F. Jakub and B. Mayer

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures



Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



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

Listing 1. Incomplete LU preconditioning.

Coupling of the TenStream solver to UCLA–LES

F. Jakub and B. Mayer

[Title Page](#)[Abstract](#)[Introduction](#)[Conclusions](#)[References](#)[Tables](#)[Figures](#)[◀](#)[▶](#)[◀](#)[▶](#)[Back](#)[Close](#)[Full Screen / Esc](#)[Printer-friendly Version](#)[Interactive Discussion](#)

```
-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
```

Listing 2. Multigrid preconditioning.