P R E P R I N T
April 13, 2011

AN ADAPTIVE DISCONTINUOUS GALERKIN METHOD FOR MODELING ATMOSPHERIC CONVECTION

Andreas Müller*, Jörn Behrens†, Francis X. Giraldo†† and Volkmar Wirth*

*Institute for Atmospheric Physics,
University of Mainz (Germany)
e-mail: andrmue@uni-mainz.de

†KlimaCampus
University of Hamburg (Germany)
e-mail: joern.behrens@zmaw.de

††Department of Applied Mathematics
Naval Postgraduate School
Monterey, California (USA)
e-mail: fxgirald@nps.edu

Key words: Discontinuous Galerkin, Large Eddy Simulation, Adaptive, Meteorology, Clouds, Semi-Implicit

Abstract. *Theoretical understanding and numerical modeling of atmospheric moist convection still pose great challenges to meteorological research. A Direct Numerical Simulation of a single cumulus cloud is beyond the capacity of today's computing power. The use of a Large Eddy Simulation in combination with semi-implicit time-integration and adaptive techniques offers a significant reduction of complexity.*

This paper presents a first step towards an efficient simulation of a single cloud. So far this work is restricted to dry flow in two-dimensional geometry without subgrid-scale modeling. The compressible Euler equations are discretized using a discontinuous Galerkin method introduced by Giraldo and Warburton in 2008. Time integration is done by a semi-implicit backward difference formula. This paper represents the first application of a triangular discontinuous Galerkin method with h-adaptive grid refinement for nonhydrostatic atmospheric flow. This refinement of our triangular grid is implemented with the function library AMATOS and uses an efficient space filling curve approach.

Validation through different test cases shows very good agreement between the current results and those from the literature. For comparing different adaptivity setups we developed a new qualitative error measure for the simulation of warm air bubbles. With the help of this criterion we show that the simulation of a rising warm air bubble on a locally refined grid can be four times faster than a similar computation on a uniform mesh while still producing the same accuracy. Remarkably only 5% of the total CPU time is used for adapting the grid after each time-step.

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
13 APR 2011		**00-00-2011 to 00-00-2011**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
An Adaptive Discontinuous Galerkin Method for Modeling Atmospheric Convection	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Naval Postgraduate School,Department of Applied Mathematics,Monterey,CA,93943**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Theoretical understanding and numerical modeling of atmospheric moist convection still pose great challenges to meteorological research. A Direct Numerical Simulation of a single cumulus cloud is beyond the capacity of today?s computing power. The use of a Large Eddy Simulation in combination with semi-implicit time-integration and adaptive techniques offers a significant reduction of complexity. This paper presents a first step towards an efficient simulation of a single cloud. So far this work is restricted to dry flow in two-dimensional geometry without subgrid-scale modeling. The compressible Euler equations are discretized using a discontinuous Galerkin method introduced by Giraldo and Warburton in 2008. Time integration is done by a semi-implicit backward difference formula. This paper represents the first application of a triangular discontinuous Galerkin method with h-adaptive grid refinement for nonhydrostatic atmospheric flow. This refinement of our triangular grid is implemented with the function library AMATOS and uses an efficient space filling curve approach. Validation through different test cases shows very good agreement between the current results and those from the literature. For comparing different adaptivity setups we developed a new qualitative error measure for the simulation of warm air bubbles. With the help of this criterion we show that the simulation of a rising warm air bubble on a locally refined grid can be four times faster than a similar computation on a uniform mesh while still producing the same accuracy. Remarkably only 5% of the total CPU time is used for adapting the grid after each time-step.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**20**	

1 INTRODUCTION

Significant progress in numerous areas of scientific computing comes from the steadily increasing capacity of computers and the advances in numerical methods. An example is the simulation of the Earth's atmosphere, which has proven to be extremely challenging owing to its multiscale and multi-process nature. Even with todays computers it is impossible to explicitly represent all scales and all processes involved. To overcome this difficulty one resorts to empirically-based closure approaches — called "parameterisations" — that try to capture the unresolved aspects of the problem. Needless to say, this introduces errors.

An application with high practical relevance is numerical weather prediction. Generally, its skill has improved considerably over the past few decades, and a significant portion of this improvement has been attributed to the increased computing power and refined numerical methods[1]. A notable exception to this general development is the forecast of precipitation, where the progress has been almost non-existent[2]. The reason for this state of affairs is likely due to the fact that most processes leading to precipitation are parameterized rather than explicitly simulated in today's weather prediction models. In this paper we shall develop and present a new numerical model that is specifically tailored to investigate one of these processes in detail.

The paper is organized as follows. First, in section 2 we motivate the new model development by explaining the meteorological problem. In section 3 we then present the numerical methods used in our work. This includes the discontinuous Galerkin method for the spatial discretization, a semi-implicit method for the time integration, and a space filling curve approach for the adaptive grid management. In section 4 we apply our code to three test cases from the literature. Section 5 provides some tests concerning the accuracy of the adaptive mesh refinement. The paper ends with a summary and outlook in section 6.

2 METEOROLOGICAL MOTIVATION

A single cumulus cloud can be considered as a prototypical basic element of atmospheric moist convection. The cloud rises through the environmental air owing to its positive buoyancy (fig. 1). Upward motion of the cloud (thick blue arrow) is associated with downward motion in a thin shell surrounding the rising cloud (thin blue arrows)[3]. This induces wind shear at the cloud-environment interface, leading to Kelvin-Helmholtz instabilities which eventually result in turbulence. The ensuing mixing between moist cloudy and dry environmental air leads to evaporation of cloud droplets. This cools the parcel resulting in negative buoyancy corresponding to a downward force (red arrows). This process is aptly called "buoyancy reversal"[5,6,7].

Early indications for the significance of buoyancy reversal for cloud dynamics stem from the laboratory experiments of Johari[8]. He introduced buoyancy reversal in his watertank experiments with the help of chemical reactions occurring in the mixing region between

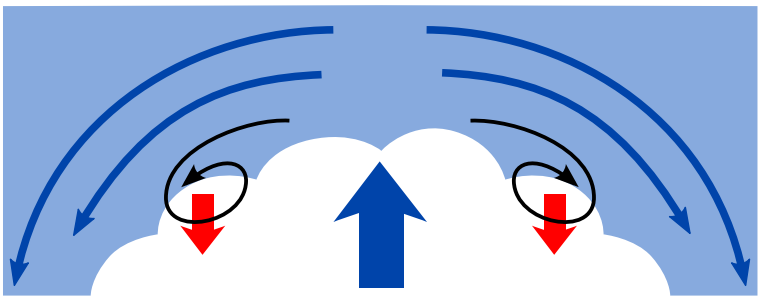

Figure 1: Illustration of buoyancy reversal. The blue arrows demonstrate the mean flow of a rising cloud, the black arrows represent turbulence produced by Kelvin-Helmholtz instability and the red arrows illustrate buoyancy reversal. For further explanation we refer to the text.

two fluids. Johari found that, depending on the strength of the buoyancy reversal, the morphology of the cloud development could be vastly different.

Similar results were found in highly idealized numerical two-fluid experiments by Grabowski[4] in 1995. These simulations started with two fluid layers, one on top of the other. Convergence was imposed in the lower layer thus leading to a rising plume. The simulation was done twice: with and without buoyancy reversal. The buoyancy reversal was implemented as an additional downward force at the interface between the two fluids. Initially the two simulations looked very similar but, after further development, increasing differences could be seen.

These preliminary investigations suggest that buoyancy reversal has an important impact on cloud dynamics and, hence, on the formation of precipitation. However, owing to their idealized nature it is not possible to draw any firm conclusion about real clouds. On the other hand, numerical weather prediction models and even so-called "cloud resolving models" are not able to explicity simulate the processes relevant for buoyancy reversal due to their coarse spatial resolution[10]. It is here that we want to take a step forward by developing a new numerical model that is specifically designed to deal with the mixing processes at the cloud boundary.

A Direct Numerical Simulation (DNS) would require a resolution of about 1 mm in each direction in order to properly resolve all dynamical scales[10]. In three dimensions this amounts to some 10^{24} grid points, which is beyond the capacity of today's computing power. We, therefore, resort to Large Eddy Simulation (LES) in combination with a semi-implicit time integration method. The use of an adaptive technique will offer a significant reduction in numerical expense, as it allows us to focus attention to the cloud-environment interface, which is the region where mixing and buoyancy reversal takes place.

3 NUMERICAL METHODS

In this section we present the numerical methods that are used in our work. The choice of the numerical method requires some *a priori* knowledge of the application for which the numerical model is designed. Measurements in real clouds have shown a large variety of behavior[9]: there are clouds with numerous steep gradients in the interior (see for example the liquid water content in figure 15 of that reference[9]). On the other hand, smaller clouds often have a fairly smooth interior with discontinuities mostly at the boundary of the cloud (fig. 13 of Damiani et al.[9]). It is the latter which we intend to simulate. For such clouds a discontinuous Galerkin (DG) discretization in combination with a semi-implicit time-integrator should be the best choice; the reason for this choice is due to the high-order accuracy and robustness in handling discontinuities of the DG method as well as the large time-steps allowed by the semi-implicit method. For avoiding artificial oscillations an artificial viscosity is used. Before describing these methods in the following subsections we first introduce the equations that we use in our simulations.

In the current work the fully compressible Euler equations are used. For the dry case we use the following set of equations (see equation set 2 in Giraldo and Restelli[13]):

$$\frac{\partial \rho}{\partial t} + \boldsymbol{\nabla} \cdot (\rho\, \boldsymbol{u}) = 0, \tag{1}$$

$$\frac{\partial \rho \boldsymbol{u}}{\partial t} + \boldsymbol{\nabla} \cdot (\rho\, \boldsymbol{u} \otimes \boldsymbol{u} + p\, I_2) = -\rho\, g\, \hat{\mathbf{k}}, \tag{2}$$

$$\frac{\partial \rho\theta}{\partial t} + \boldsymbol{\nabla} \cdot (\rho\, \theta\, \boldsymbol{u}) = 0, \tag{3}$$

where the variables are $(\rho, \rho\boldsymbol{u}, \rho\theta)^{\mathrm{T}}$, ρ is the density, $\boldsymbol{u} = (u, w)^{\mathrm{T}}$ is the velocity field and θ is the potential temperature. Furthermore we denote the gravitational constant with g, the tensor product by $\otimes$, the identity matrix in $\mathbb{R}^2$ by I_2, the unit vector in the vertical direction with $\hat{\mathbf{k}}$.

Pressure p in eq. (2) is given by the equation of state:

$$p = p_0 \left(\frac{\rho\, R\, \theta}{p_0} \right)^{\frac{c_p}{c_v}}, \tag{4}$$

with a constant reference pressure p_0, the gas constant $R = c_p - c_v$ and the specific heats for constant pressure and volume, c_p and c_v. Potential temperature θ is defined by

$$\theta = T \left(\frac{p_0}{p} \right)^{\frac{R}{c_p}} \tag{5}$$

with temperature T. This definition can be illustrated in the following way: if we consider dry air with temperature T and pressure p, potential temperature θ is given by the temperature which the air would have when being transported adiabatically to a place

with pressure p_0. In our model we use potential temperature as a variable because this simplifies the extension to moist air in future research.

Atmospheric flow is often approximately in hydrostatic balance, which is defined by

$$\frac{\partial p}{\partial z} = -\rho\, g. \tag{6}$$

This balance can produce numerical instabilities because the remaining terms in the vertical component of eq. 2 are much smaller than the two terms of the hydrostatic balance (6). To avoid this instability we introduce the mean states $\bar{p}$, $\bar{\rho}$ and $\bar{\theta}$ that are in hydrostatic balance. The mean state of pressure p is defined by $\bar{p} = p(\bar{\rho}, \bar{\theta})$. The deviation of the variables from the mean state is denoted by $\rho' = \rho - \bar{\rho}$, $\theta' = \theta - \bar{\theta}$ and $p' = p - \bar{p}$. By this procedure the set of equations (1) – (3) can be written as

$$\frac{\partial \rho'}{\partial t} + \boldsymbol{\nabla} \cdot (\rho\, \boldsymbol{u}) = 0, \tag{7}$$

$$\frac{\partial \rho \boldsymbol{u}}{\partial t} + \boldsymbol{\nabla} \cdot (\rho\, \boldsymbol{u} \otimes \boldsymbol{u} + p'\, I_2) = -\rho'\, g\, \hat{\mathbf{k}}, \tag{8}$$

$$\frac{\partial \rho \theta'}{\partial t} + \boldsymbol{\nabla} \cdot (\rho\, \theta\, \boldsymbol{u}) = 0. \tag{9}$$

To discretize these equations in space we introduce the commonly used notation

$$\frac{\partial \boldsymbol{q}}{\partial t} + \boldsymbol{\nabla} \cdot \mathbf{F}\,(\boldsymbol{q}) = \boldsymbol{S}\,(\boldsymbol{q})\,, \tag{10}$$

with the vector $\boldsymbol{q} = (\rho, \rho\boldsymbol{u}, \rho\theta)^{\mathrm{T}}$, the source function

$$\boldsymbol{S}\,(\boldsymbol{q}) = \begin{pmatrix} 0 \\ -\rho'\, g\, \hat{\mathbf{k}} \\ 0 \end{pmatrix}, \tag{11}$$

and the flux tensor

$$\mathbf{F}\,(\boldsymbol{q}) = \begin{pmatrix} \rho\, \boldsymbol{u} \\ \rho\, \boldsymbol{u} \otimes \boldsymbol{u} + p'\, I_2 \\ \rho\, \theta\, \boldsymbol{u} \end{pmatrix}. \tag{12}$$

Equation (10) is discretized using the discontinuous Galerkin method which we now describe in detail.

3.1 Discontinuous Galerkin Method

In our work we use a discontinuous Galerkin method based on the strong formulation using the Rusanov flux at the cell interfaces. Furthermore, we consider a two dimensional

triangular mesh; the extension to a full three dimensional method will remain a task for the future but we envision using tetrahedra for this task. The triangular discontinuous Galerkin method used in our work is described by Giraldo and Warburton[11] for the case of the shallow water equations. Despite a different definition of conserved variables $\boldsymbol{q}$, flux tensor $\mathbf{F}(\boldsymbol{q})$ and source function $\boldsymbol{S}(\boldsymbol{q})$, eq. (10) remains unchanged. Therefore, we repeat in this paper only the main ideas of the discretization.

We start by multiplying eq. (10) with a test function ψ, integrating over an arbitrary element Ω_e and bringing the spatial derivative in front of the test function with integration by parts. Replacing the flux in the boundary terms by a numerical flux $\mathbf{F}^*$ leads to the following equation for the numerical solution $\boldsymbol{q}_N$:

$$\int_{\Omega_e} \left(\frac{\partial \boldsymbol{q}_N}{\partial t} - \mathbf{F}_N \cdot \boldsymbol{\nabla} - \boldsymbol{S}_N \right) \psi(\boldsymbol{x}) \, \mathrm{d}\mathbf{x} = - \int_{\Gamma_e} \psi(\boldsymbol{x}) \, \hat{\mathbf{n}} \cdot \mathbf{F}_N^* \mathrm{d}\mathbf{x}, \tag{13}$$

where Γ_e is the boundary of element Ω_e, $\hat{\mathbf{n}}$ is the outward pointing unit normal vector on Γ_e, $\mathbf{F}_N = \mathbf{F}(\boldsymbol{q}_N)$ and $\boldsymbol{S}_N = \boldsymbol{S}(\boldsymbol{q}_N)$. Applying again integration by parts gives the *strong formulation*[12]

$$\int_{\Omega_e} \left(\frac{\partial \boldsymbol{q}_N}{\partial t} + \boldsymbol{\nabla} \cdot \mathbf{F}_N - \boldsymbol{S}_N \right) \psi(\boldsymbol{x}) \, \mathrm{d}\mathbf{x} = \int_{\Gamma_e} \psi(\boldsymbol{x}) \, \hat{\mathbf{n}} \cdot (\mathbf{F}_N - \mathbf{F}_N^*) \, \mathrm{d}\mathbf{x}. \tag{14}$$

Now we introduce an expansion by polynomial basis functions

$$\boldsymbol{q}_N(\boldsymbol{x}) = \sum_{j=1}^{M_N} \psi_j(\boldsymbol{x}) \, \boldsymbol{q}_j \tag{15}$$

and assume that the test function ψ can be written as a linear combination of the basis functions. Using Einstein's sum convention we get

$$\frac{\partial \boldsymbol{q}_i}{\partial t} = - \int_{\Omega_e} \hat{\psi}_i \left(\boldsymbol{\nabla} \cdot \mathbf{F}_N - \boldsymbol{S}_N \right) \mathrm{d}\mathbf{x} + \int_{\Gamma_e} \hat{\psi}_i \hat{\mathbf{n}} \, \mathrm{d}\mathbf{x} \cdot \left(\mathbf{F}_N - \mathbf{F}_N^* \right), \tag{16}$$

where $\hat{\psi}_i = M_{ik}^{-1} \psi_k$ with the mass matrix $M_{ik} = \int_{\Omega_e} \psi_i \psi_k \mathrm{d}\mathbf{x}$. For the sake of simplicity, we did not write the dependence on $\boldsymbol{x}$ of the basis functions although it should be understood that the basis functions depend on the spatial coordinates.

The integrals in eq. (16) are evaluated using high order quadrature rules[11]. F^* is discretized as the Rusanov flux, given by:

$$\mathbf{F}_N^* = \frac{1}{2} \left[\mathbf{F}\left(\boldsymbol{q}_N^L\right) + \mathbf{F}\left(\boldsymbol{q}_N^R\right) - \lambda \, \hat{\mathbf{n}} \left(\boldsymbol{q}_N^R - \boldsymbol{q}_N^L \right) \right] \tag{17}$$

with the maximum wave speed $\lambda = \max(|u| + a, |u| - a)$ where a is the speed of sound. If the normal vector $\hat{\mathbf{n}}$ of element Ω_e is pointing to the right, $\boldsymbol{q}_N^L$ is the left limiting value of $\boldsymbol{q}_N$ and $\boldsymbol{q}_N^R$ is the right limiting value.

So far we have derived a discontinuous Galerkin discretization for our set of equations (10). At this point, the right hand side of eq. (16) is known and we can integrate the equation in time. This can be done either by an explicit or implicit method. For an explicit method we implement a third order Runge-Kutta method of Cockburn and Shu [15]. Because of the fast sound and gravity waves this explicit time-integration is restricted to a very short time-step. As explained before we are not interested in simulating these fast waves accurately; therefore, we also use a semi-implicit time-integrator as presented in the next subsection.

3.2 Semi-Implicit Time Integration

The semi-implicit time integration is implemented in a similar fashion to the approach of Restelli and Giraldo [16,17]. The main difference is that we use potential temperature instead of total energy as the fourth variable.

The full nonlinear operator $\mathcal{N}(\boldsymbol{q})$ is given in our notation by

$$\mathcal{N}(\boldsymbol{q}) = -\boldsymbol{\nabla} \cdot \mathbf{F}(\boldsymbol{q}) + \boldsymbol{S}(\boldsymbol{q}). \tag{18}$$

For the semi-implicit approach we define an operator $\mathcal{L}$ by

$$\mathcal{L}\boldsymbol{q} = \begin{pmatrix} \boldsymbol{\nabla} \cdot (\rho\, \boldsymbol{u}) \\ \partial \tilde{p}'/\partial x \\ \partial \tilde{p}'/\partial z + g\, \rho' \\ \boldsymbol{\nabla} \cdot \left(\bar{\theta}\, \rho\, \boldsymbol{u}\right) \end{pmatrix}, \tag{19}$$

where $\tilde{p}'$ is given by $\tilde{p}' = \frac{c_p\, \bar{p}}{c_v\, \bar{\rho}\, \bar{\theta}}\, \rho\, \theta'$. The operator $\mathcal{L}$ is linear in the conserved variables $(\rho, \rho\, \boldsymbol{u}, \rho\, \theta)$ and $\tilde{p}'$ is a linearized version of p' with respect to the conserved variables. As explained by Restelli [19] the operator $\mathcal{L}$ is responsible for the fast moving sound and gravity waves and, therefore, must be integrated implicitly. This splitting is done by writing

$$\frac{\partial \boldsymbol{q}}{\partial t} = \{\mathcal{N}(\boldsymbol{q}) - \mathcal{L}\,\boldsymbol{q}\} + \mathcal{L}\,\boldsymbol{q}. \tag{20}$$

For discretizing (20) in time, we use a backward difference formula of order 2, that leads to

$$\frac{1}{\gamma\, \Delta t} \sum_{m=-1}^{1} \alpha_m \boldsymbol{q}^{n-m} = \sum_{m=0}^{1} \beta_m \left[\mathcal{N}\left(\boldsymbol{q}^{n-m}\right) - \mathcal{L}\,\boldsymbol{q}^{n-m}\right] + \mathcal{L}\,\boldsymbol{q}^{n+1} \tag{21}$$

with $\alpha_{-1} = 1$, $\alpha_0 = 4/3$, $\alpha_1 = -1/3$, $\gamma = 2/3$, $\beta_0 = 2$ and $\beta_1 = -1$. We rewrite this equation collecting all terms with $\boldsymbol{q}^{n+1}$ and get

$$\left[1 - \gamma\, \Delta t\, \mathcal{L}\right]\, \boldsymbol{q}^{n+1} = \tilde{\boldsymbol{q}}^{\mathrm{ex}} - \gamma\, \Delta t \sum_{m=0}^{1} \beta_m \mathcal{L}\, \boldsymbol{q}^{n-m}, \tag{22}$$

where

$$\tilde{\boldsymbol{q}}^{\mathrm{ex}} = \sum_{m=0}^{1} \alpha_m \boldsymbol{q}^{n-m} + \gamma\, \Delta t \sum_{m=0}^{1} \beta_m \mathcal{N}\left(\boldsymbol{q}^{n-m}\right) \tag{23}$$

is an explicit predictor that has to be calculated first. Solving the linear system of equations (22) (e.g., with GMRES) gives the implicit corrector. For details on this solution strategy the reader is referred to[17] and[18].

3.3 Mesh Refinement with Space Filling Curve Approach

As explained in section 2 we expect steep gradients at the boundary of the cloud. For increasing the numerical resolution in these regions we use h-adaptive mesh refinement. This is managed with the function library AMATOS[23]. The main advantage of this function library is that it handles the entire h-adaptive mesh refinement. Furthermore it orders the unknowns very efficiently by using a so-called space filling curve approach. For further information we refer to the publication of Behrens et al.[23].

The only modification that was necessary for our work was the calculation of the new values at the grid points when elements are refined or coarsened. This is quite straight forward. We simply evaluate the old polynomials at the positions of the new degrees of freedom.

The refinement criterion used for the results presented in this paper is given by:

$$|\theta'(\boldsymbol{x}, t)| \geq \max_{\boldsymbol{x}} \left(|\theta'(\boldsymbol{x}, t)|\right) / 10. \tag{24}$$

Wherever this condition is fulfilled the mesh is refined until it reaches a specified finest resolution. In the rest of the domain the grid is coarsened until it reaches a specified coarsest resolution without modifying the resolution in the refinement region. The transition between fine and coarse meshes is given by the conformity of the grid. To avoid small scale structures moving into a region with a coarse mesh we add two to four rows of fine elements to the refinement region. For more realistic simulations, we intend to utilize more involved refinement criteria, which consider physical behavior, gradients, and other indicators.

3.4 Artificial viscosity

For avoiding artificial oscillations we use a so called artificial viscosity. This is done by adding a diffusion term $\boldsymbol{\nabla} \cdot (\mu\, \rho\, \boldsymbol{\nabla} \boldsymbol{u})$ to the right hand side of eq. (2) and $\boldsymbol{\nabla} \cdot (\mu\, \rho\, \boldsymbol{\nabla} \theta')$ to the right hand side of eq. (3) with the viscosity parameter μ. These terms represent the simplest variational multiscale approach one could use.

For discretizing the second order derivates produced by the artificial viscosity we use a local discontinuous Galerkin method[14]. That means that the order of the derivatives is

reduced by introducing the following new variables

$$\boldsymbol{\alpha} = \boldsymbol{\nabla} \boldsymbol{u}, \tag{25}$$

$$\beta = \boldsymbol{\nabla} \theta. \tag{26}$$

Similar to eq. (16) we get

$$\boldsymbol{\alpha}_i = -\int_{\Omega_e} \hat{\psi}_i \, \boldsymbol{u}_j \boldsymbol{\nabla} \psi_j \mathrm{d}\mathbf{x} + \int_{\Gamma_e} \hat{\psi}_i \psi_j \hat{\mathbf{n}} \, \mathrm{d}\mathbf{x} \left(\boldsymbol{u}_j - \boldsymbol{u}_j^* \right), \tag{27}$$

$$\beta_i = -\int_{\Omega_e} \hat{\psi}_i \, \theta_j \boldsymbol{\nabla} \psi_j \mathrm{d}\mathbf{x} + \int_{\Gamma_e} \hat{\psi}_i \psi_j \hat{\mathbf{n}} \, \mathrm{d}\mathbf{x} \left(\theta_j - \theta_j^* \right). \tag{28}$$

As these viscosity terms do not describe a flow in a certain direction (as in the case of the advection terms) we use the following numerical flux for the viscosity terms in $\mathbf{F}_N^*$ as well as for $\boldsymbol{u}^*$ and θ^*:

$$\mathbf{F}_{\text{visc}}^* \left(\boldsymbol{q}_N \right) = \frac{1}{2} \left(\mathbf{F} \left(\boldsymbol{q}_N^R \right) + \mathbf{F} \left(\boldsymbol{q}_N^L \right) \right). \tag{29}$$

Note that this is not the only possibility for discretizing the second order operators; for other choices see Shahbazi et al.[20].

For the density current test case of Straka et al.[22] we use the viscosity parameter of $\mu = 75\mathrm{m}^2/\mathrm{s}$ as defined by Straka et al.[22]. The other results presented in this paper are obtained with a viscosity parameter of $\mu = 0.1\mathrm{m}^2/\mathrm{s}$. We found that this artificial viscosity is very well suited for avoiding artificial oscillations for the numerical resolutions used in this paper.

4 VALIDATION

For the validation of our new numerical model we considered three test cases that are relevant for atmospheric convection. These test cases are a small cold air bubble on top of a large warm air bubble from Robert[21], a density current from Straka et al.[22], and a smooth warm air bubble from Giraldo and Restelli[13]. In these cases no exact solution exists but we can compare our results with those from the literature.

4.1 Small Cold Air Bubble on Top of Large Warm Air Bubble

The first test case that we consider is a small cold air bubble on top of a large warm air bubble in a domain of 1km×1km. This test case was introduced by Robert[21] in 1993. The background state has a constant potential temperature of $\bar{\theta} = 300\,\mathrm{K}$. Both bubbles have a Gaussian profile in θ'. The warm air bubble has an amplitude of 0.5 K, the amplitude of the cold air bubble is 0.17 K. The initial conditions are chosen identically to those in the publication of Robert[21].

In our test, however, we introduce two modifications. First, we use a slightly different resolution; the shortest element edge in our simulation has a length of 11m. In combination

with third order polynomials the number of degrees of freedom corresponds to a first order method with a resolution of about 4m. We call this reduced value "effective resolution". With 4m it is slightly smaller than 5m of Robert[21].

Figure 2 shows our result for this test case. The mesh is continuously adapted to the position of the temperature anomaly. For this reason the fine mesh follows the bubbles very nicely. By comparing our result with the corresponding figure of Robert[21] one can see that the results agree very well. After 600s the position and shape of the warm air is still almost identical to the corresponding plot of Robert[21]. Even the smaller vortices on the right hand side of the domain are very similar to the result of Robert.

4.2 Density Current

A second test case is a density current initialized by a cold air bubble with a cosine profile and an amplitude of 15 K in θ' (figure 3). This test case was introduced by Straka et al.[22]. The viscosity of $\mu = 75\text{m}^2/\text{s}$ is identical to the setup of Straka et al.[22]. As in the previous test case we use third order polynomials. The shortest element edge has a length of 50m in our computation. This leads again to an effective resolution that is slightly smaller than the smallest resolution of Straka et al. with 25m. Again we see no differences between our result and the results in the literature. The position of the density current and the shape of the Kelvin-Helmholtz rotors agrees very well with the result of Straka et al.[22] throughout the whole simulation.

4.3 Smooth Warm Air Bubble

As a third test case we computed the rising thermal bubble introduced by Giraldo and Restelli[13] (test case 2) in 2008. It is a single warm air bubble with a cosine profile in θ'. As in the test case of Robert[21] the domain has an extent of 1km in each direction and the bubble has an amplitude of 0.5 K. We use the same parameters as in the publication of Giraldo and Restelli[13] except an additional artificial viscosity and a modified numerical resolution. In our case the shortest element edge has a length of 11m. By using third order polynomials our effective resolution is about 4m. This is significantly larger than all the effective resolutions used by Giraldo and Restelli. As they use 10th order polynomials one has to divide their resolution by a factor of three to get a comparable number of degrees of freedom as in our simulation. This explains that they get almost no artificial oscillations even without using artificial viscosity.

As in the previous test cases there are no obvious differences between our results and those from the literature. The position and shape of the bubble seems to be identical to the result of Giraldo and Restelli[13]. This gives us confidence that our code is error-free.

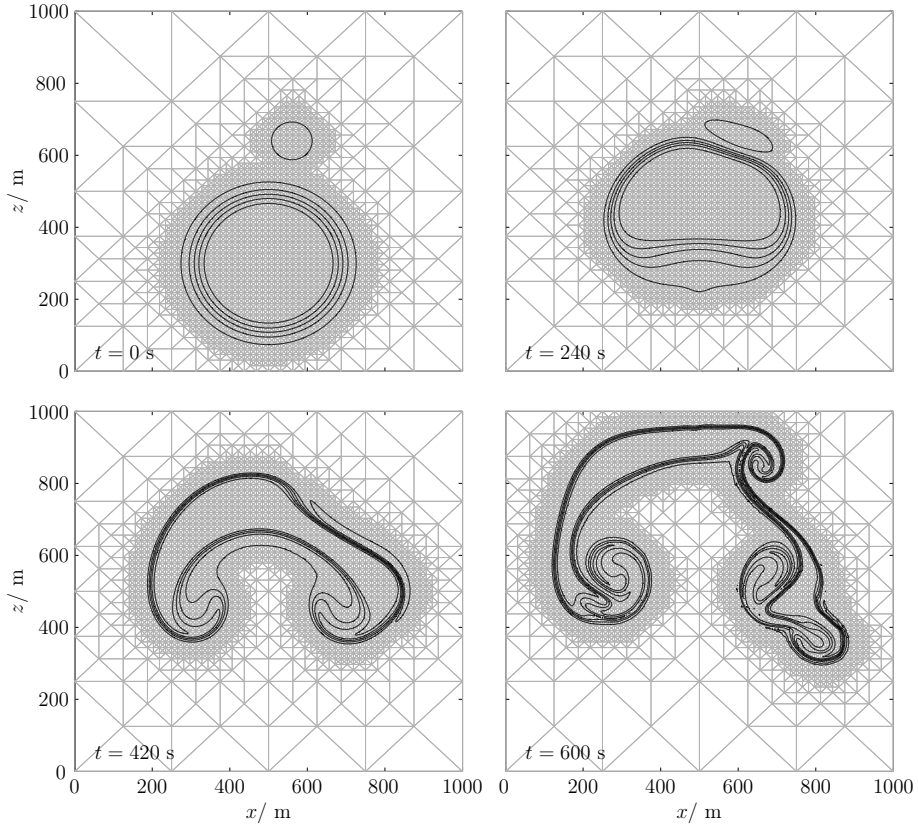

Figure 2: Small cold air bubble on top of a large warm air bubble as introduced by Robert[21]. The contour lines show the deviation of the potential temperature from the background state and the gray lines show the adaptively refined triangular mesh used in our simulation. The contour values are from -0.05 K to 0.45 K with an interval of 0.05 K. For avoiding artificial oscillations we use a constant artificial viscosity of $\mu = 0.1\mathrm{m}^2/\mathrm{s}$. For the time-integration we used the explicit Runge-Kutta method. Semi-implicit time-integration produces the same results.

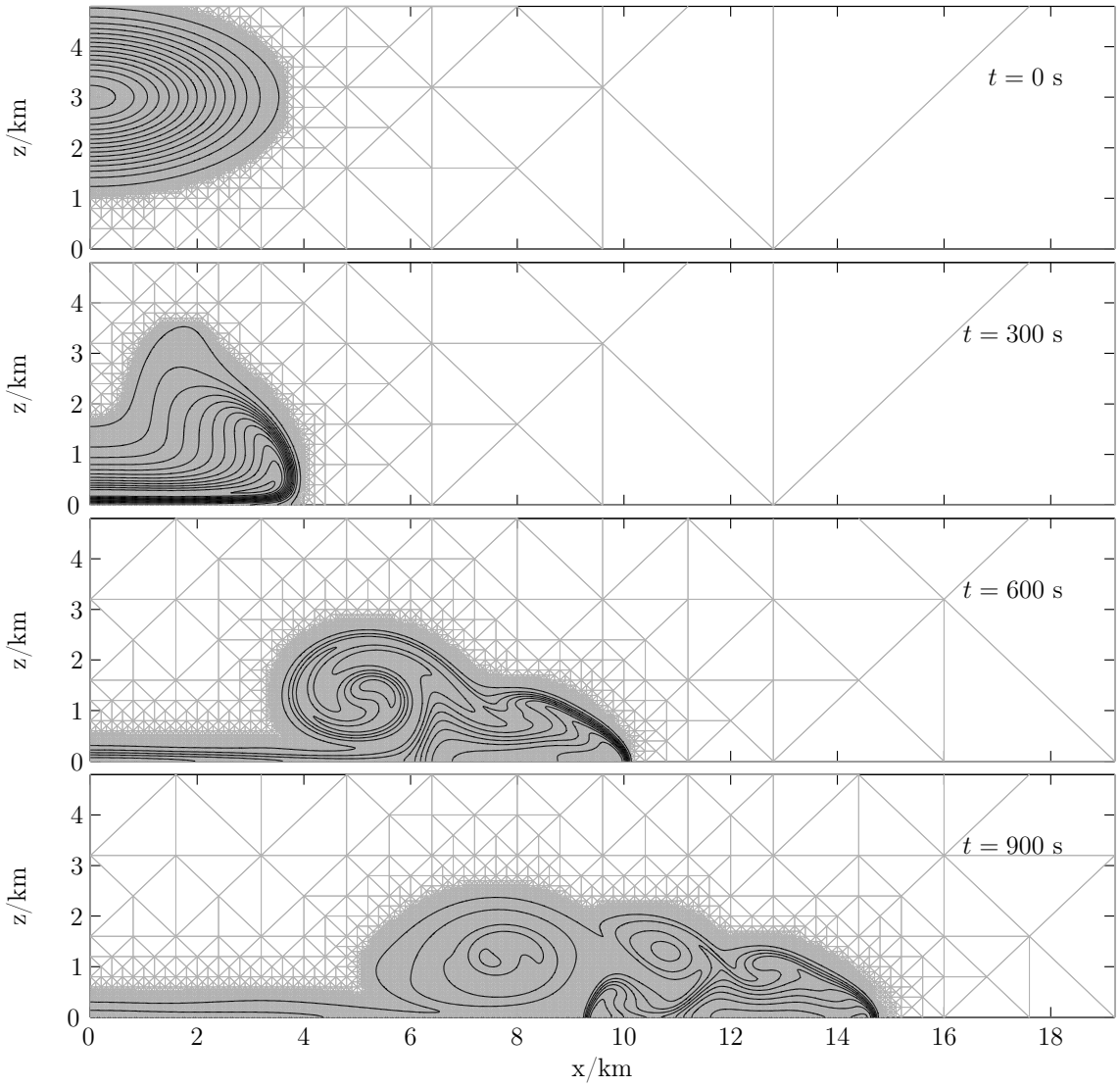

Figure 3: Density current as introduced by Straka et al.[22]. As in figure 2 the contour lines show the deviation θ' of the potential temperature θ from the background state $\bar{\theta}$ and gray lines show the adaptively refined triangular mesh. Contour values are from 0.5 K to 14.5 K with an interval of 1 K. Because of the fairly strong viscosity of $\mu = 75\text{m}^2/\text{s}$ no values outside this range exist.

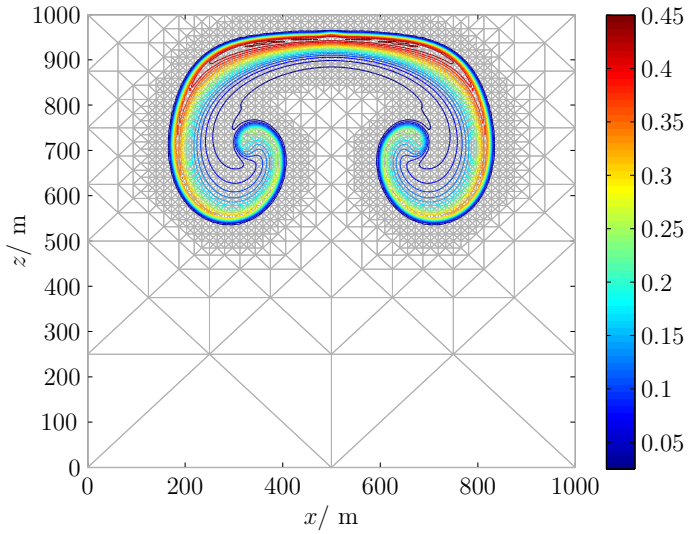

Figure 4: Rising thermal bubble introduced by Giraldo and Restelli[13]. As in figure 2 the contour lines show the deviation θ' of the potential temperature θ from the background state $\bar{\theta}$ and gray lines show the adaptively refined triangular mesh. Contour values are from 0.025 K to 0.45 K with an interval of 0.025 K. Because of the artificial viscosity of $\mu = 0.1\text{m}^2/\text{s}$ there exist no negative values and no values larger than 0.45 K.

5 SENSITIVITY STUDIES

One important question for each adaptive numerical model is: how accurate is the adaptive method? For comparing different adaptivity setups we introduce a new error measure for the simulation of warm air bubbles. In this section we start with describing this criterion that is used later for some sensitivity studies. These studies include a comparison between a simulation on an adaptive mesh with a simulation on a uniform mesh and a sensitivity study concerning the size of the refinement region.

5.1 Comparison Criterion

For comparing different adaptivity setups we had to find a criterion that gives at least an approximation of the accuracy of an adaptive simulation. As explained before no exact solution for the warm air bubble test case is known. Hence it is impossible to calculate the error of those simulations. Furthermore the results for increasing numerical resolution do not converge as more and more turbulence is resolved. Instead we sought a qualitative measure for the numerical errors. For this purpose we use the rising thermal bubble of Giraldo and Restelli[13] as in section 4.3. But this time we continue the simulation much longer until numerical errors become clearly visible. After 900 seconds we get the results shown in figure 5 for three different resolutions with a constant artificial viscosity of $\mu = 0.1\mathrm{m}^2/\mathrm{s}$. This figure suggests, that the results converge. Furthermore one can see that vortices which arise due to the Kelvin-Helmholtz instabilities at the left bottom part of the bubble strongly depend on numerical resolution. This is not surprising. A similar situation can be seen in the case of shear-instability of a horizontal shear-flow. The exact solution of a perfectly horizontal shear-flow is just stationary[24]. A perfectly horizontal shear-flow will remain perfectly horizontal as long as there is no perturbation. Some kind of perturbation is necessary for making the instability of the flow visible. In a similar way we expect that a perfectly circular warm air bubble in an infinitely large domain should never develop Kelvin-Helmholtz instabilities. But even a tiny perturbation will grow in time and becomes visible for a long enough simulation time. In our numerical simulations the perturbation is given only by numerical errors. So we can compare the accuracy of different simulations by comparing the time when the instability at the bottom left side of the bubble develops.

This criterion is not suitable for comparing different numerical models. The onset of Kelvin-Helmholtz instability depends also on diffusion. The larger the diffusion the later the instability can be seen. A numerical model with strong numerical diffusion shows less instability but is not more accurate. Therefore the artificial viscosity has to be large compared to possible differences in the numerical diffusion of the simulations that are compared.

A further remark concerning figure 5. At the top left corner of the domain, one can see a vortex that is almost independent of the numerical resolution. This vortex is caused by Kelvin-Helmholtz instability too, but in this case the instability is initiated by the solid

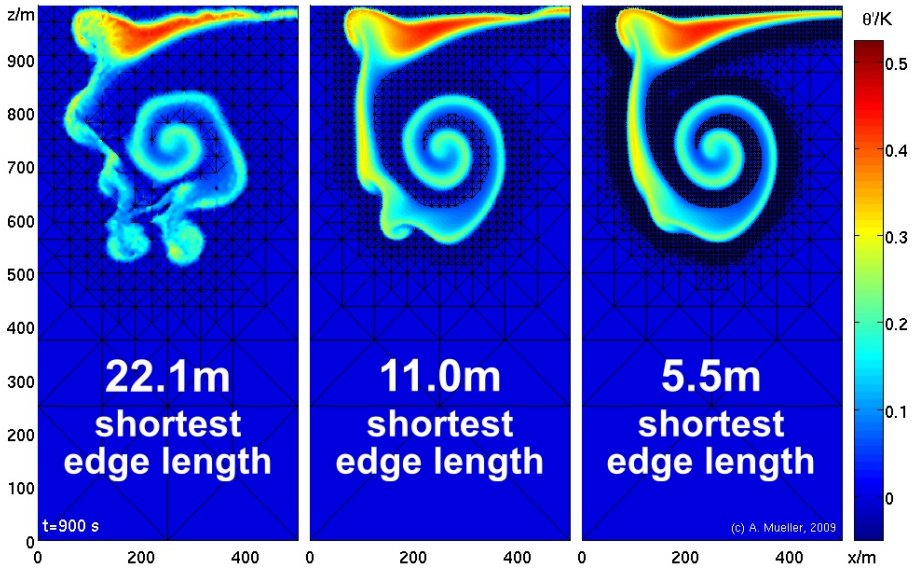

Figure 5: Rising thermal bubble as in figure 4 for three different resolutions with a constant artificial viscosity of $\mu = 0.1\text{m}^2/\text{s}$. Shown is only the left half of the bubble. Color shading indicates the potential temperature deviation from the background state. The adaptively refined triangular mesh is shown by the black lines. The mesh is not shown where the refinement criterion (24) is fulfilled. The fine grid surrounding this region has to be continued in this refinement region. The numerical resolution is given by the length of the shortest element edge.

wall boundaries. A similar behavior can be seen in the test case of Straka et al. in figure 3. These vortices are almost independent of the numerical resolution and cannot be used for comparing the accuracy of different simulations.

5.2 Adaptive versus Uniform

With the criterion introduced in the previous subsection we now compare a simulation performed on an adaptively refined triangular mesh with a simulation on a uniform grid. The result is shown in figure 6. The onset of the Kelvin-Helmholtz instability and even the small vortices produced by the instability are almost identical in both simulations. This shows that the locally refined mesh is able to produce the same accuracy as a uniform mesh.

But there is one important difference between these two simulations: the simulation on the locally refined mesh is four times faster than the simulation using the uniform mesh (see table 1). Only 5% of the CPU time is used for adapting the grid and only one quarter of the elements is used, i.e. 25% memory requirements compared to the uniform grid. This demonstrates very clearly the advantage of using adaptive mesh refinement for the simulation of atmospheric convection.

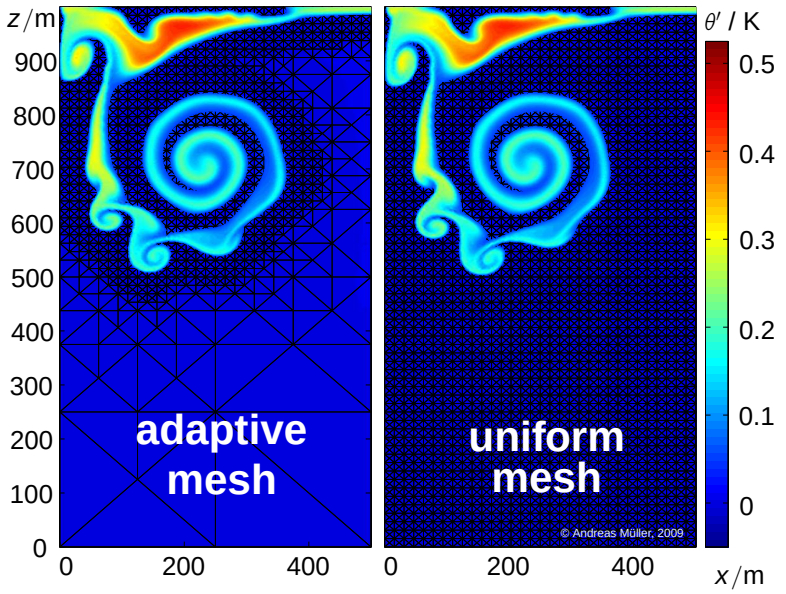

Figure 6: Comparison between a simulation using an adaptive mesh and a simulation using a uniform mesh after 1000 seconds. The resolution of the uniform mesh is equal to the finest resolution of the adaptive mesh. The length of the shortest element edges is 11m in both cases. All parameters of the rising bubble are identical to those in figure 4.

5.3 Size of Refinement Region

As shown in the previous subsection the simulation using adaptive mesh refinement is much faster than a simulation using a uniform mesh. Nevertheless both simulations have the same accuracy. Can we improve the efficiency by using a smaller refinement region without losing accuracy? To answer this question we repeated the adaptive simulation of figure 6 with three slightly different refinement regions (figure 7).

The onset of Kelvin-Helmholtz instability is significantly earlier when reducing the size of the refinement region. This indicates that the numerical errors are increasing. In the case of this rising warm air bubble it seems to be better to use the large refinement region as in figure 6. Regardless, the computational effort is not much reduced by the smaller refinement regions. When developing new refinement criteria in future research we will have to repeat this test to ensure that the adaptive mesh gives the same accuracy as the uniform mesh.

6 SUMMARY AND OUTLOOK

In this paper we presented a numerical model for solving the stratified compressible Euler equations. It uses a high-order discontinuous Galerkin method based on triangular elements[11] in combination with a semi-implicit time-integrator[16]. This avoids the severe time-step restriction of explicit schemes. To our knowledge, this is the first time that a

	adaptive	**uniform**
number of elements:	4668	16384
grid time:	0.5h	0.0h
total time:	9.7h	38.7h

Table 1: Number of elements and CPU time for the two simulations shown in figure 6 with semi-implicit time-integration. The time values are the time used for reaching $t = 1000$s of the warm air bubble test case and the number of elements is an average value over the whole time. The grid time denotes the CPU time that was used for adapting the grid after each time step. Each simulation was done on the same single Linux CPU. The finest resolution in the adaptive simulation covers 21.7% of the whole domain.

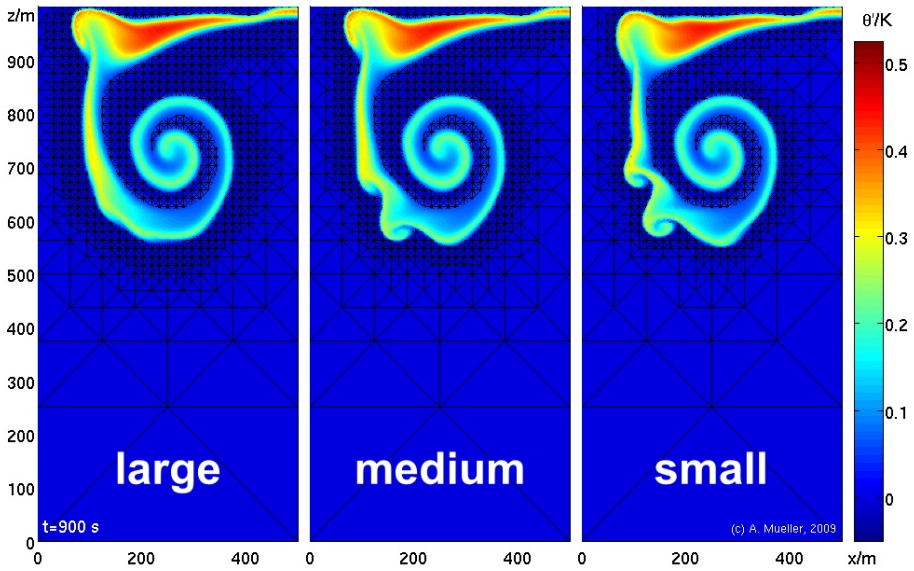

Figure 7: Adaptive simulation as in figure 6 for three different refinement regions after 900 seconds. The refinement criterion is always given by (24), but the number of additional fine elements surrounding this region is varied. From right to left: one row of fine elements is added in each simulation. The largest refinement region shown on the left panel is the same as in figure 6.

DG compressible Euler model has been combined with an h-adaptive mesh refinement for stratified flows that are of interest in nonhydrostatic meteorological applications. For the h-adaptivity we use the function library AMATOS[23] that uses a very efficient space filling curve approach.

For testing our numerical model we simulated three commonly used test cases. Our results agree well with the results from the literature.

For considering the question whether the results are affected by using adaptive mesh refinement, we developed a new qualitative error measure for the simulation of rising warm air bubbles by using artificial viscosity with a constant fairly large viscosity parameter μ. We found that Kelvin-Helmholtz instability strongly depends on the accuracy of the numerical simulation. Therefore the time when the instability becomes visible is a good criterion for comparing different simulations.

With this criterion we compared a simulation using an adaptive mesh with a simulation using a uniform mesh; the finest resolution is the same in both cases. Figure 6 shows almost no difference between the two simulations even after the instability has fully evolved. The simulation on the locally refined mesh is as accurate as the simulation on the uniform mesh. But the simulation using adaptive mesh refinement is four times faster than the simulation on the uniform mesh and only 5% of the CPU time is used for adapting the grid. This indicates that adaptive mesh refinement should be a big advantage for our future cloud simulations.

For simulating clouds we are currently working on including moisture into our model and on extending the model to 3D.

7 ACKNOWLEDGEMENT

Financial support for this work was provided by the priority program MetStröm (SPP 1276) of the german research foundation (Deutsche Forschungsgemeinschaft). Francis X. Giraldo gratefully acknowledges the support of the Office of Naval Research through program element PE-0602435N.

References

[1] E. Kalnay, S. J. Lord, and R. D. McPherson, Maturity of operational numerical weather prediction: Medium range. *Bull. Amer. Meteor. Soc*, **79**, p. 2753–2769 (1998).

[2] Deutscher Wetterdienst, Wie gut sind unsere Wettervorhersagen?, *internal report* (2002).

[3] T. Heus, and H.J.J. Jonker, Subsiding shells around shallow cumulus clouds. *J. Atmos. Sci.*, **65**, p. 1003–1018 (2008).

[4] W. W. Grabowski, Entrainment and mixing in buoyancy-reversing convection with

applications to cloud-top entrainment instability. *Quaterly Journal of the Royal Meteorological Society* **121**, 231–253 (1995).

[5] D. A. Randall, Conditional Instability of the First Kind Upside-Down. *J. Atmos. Sci.*, **37**, no. 1, p. 125–130 (1980).

[6] W. W. Grabowski, Cumulus entrainment, fine-scale mixing, and buoyancy reversal, *Quaterly Journal of the Royal Meteorological Society*, **119**, no. 513, p. 935–956 (1993).

[7] J. P. Mellado, B. Stevens, H. Schmidt, N. Peters, Buoyancy reversal in cloud-top mixing layers. *Q. J. Roy. Meteorol. Soc.*, **135**, 963–978 (2009).

[8] H. Johari, Mixing in thermals with and without buoyancy reversal. *J. Atmos. Sci.*, vol. **49**, no. 16, p. 1412–1426 (1992).

[9] R. Damiani, G. Vali, and S. Haimov, The structure of thermals in cumulus from airborne dual-Doppler radar observations. *Journal of the Atmospheric Sciences*, vol. **63**, Issue 5, pp.1432–1450 (2006).

[10] G. H. Bryan, J. C. Wyngaard, and J. M. Fritsch, Resolution requirements for the simulation of deep moist convection. *Monthly Weather Review*, vol. **131**, no. 10, p. 2394–2416 (2003).

[11] F.X. Giraldo and T. Warburton, A high-order triangular discontinuous Galerkin oceanic shallow water model, *Int. J. Numer. Methods Fl.*, Vol. 56, pp. 899–925 (2008).

[12] F.X. Giraldo, J.S. Hesthaven, and T. Warburton, A high-order triangular discontinuous Galerkin oceanic shallow water model, *Journal of Computational Physics*, Vol. 181, pp. 499-525 (2002).

[13] F.X. Giraldo and M. Restelli, A study of spectral element and discontinuous Galerkin methods for the Navier-Stokes equations in nonhydrostatic mesoscale atmospheric modeling: Equation sets and test cases, *J. Comput. Phys.*, 227, pp. 3849–3877 (2008).

[14] B. Cockburn and C-W. Shu, The local discontinuous Galerkin method for time-dependent convection-diffusion systems. *SIAM Journal on Numerical Analysis*, 35:2440–2463 (1998).

[15] B. Cockburn and C-W. Shu, Runge-Kutta discontinuous Galerkin methods for convection-dominated problems. *Journal of Scientific Computing*, 16:173–261 (2001).

[16] M. Restelli and F. Giraldo, A conservative discontinuous Galerkin semi-implicit formulation for the Navier-Stokes equations in nonhydrostatic mesoscale modeling, *Journal of Scientific Computing*, Vol. 31, No. 3, pp. 2231–2257 (2009).

[17] F.X. Giraldo and M. Restelli, High-order semi-implicit time-integrators for a triangular discontinuous Galerkin oceanic shallow water model, *Int. J. Numer. Methods Fl.*, Vol. 63, No. 9, pp. 1077-1102 (2010).

[18] F.X. Giraldo, M. Restelli, and M. Läuter, Semi-implicit formulations of the Navier-Stokes equations: application to nonhydrostatic atmospheric modeling, *SIAM Journal on Scientific Computing*, Vol. 32, No. 6, pp. 3394–3425 (2010).

[19] M. Restelli, Semi-Lagrangian and semi-implicit discontinuous Galerkin methods for atmospheric modeling applications, *Ph.D. Thesis*, Politecnico di Milano (2007).

[20] K. Shahbazi, P. F. Fischer and C. R. Ethier, A high-order discontinuous Galerkin discretization of the unsteady incompressible Navier-Stokes equations, *Journal of Computational Physics* 222, pp. 391–407 (2007).

[21] A. Robert, Bubble convection experiments with a semi-implicit formulation of the Euler equations, *Journal of the Atmospheric Sciences* 50, pp. 1865–1873 (1993).

[22] J. M. Straka, R. B. Wilhelmson, L. J. Wicker, J. R. Anderson, K. K. Droegemeier, Numerical solutions of a non-linear density current: a benchmark solution and comparisons, *International Journal for Numerical Methods in Fluids* 17, pp. 1–22 (1993).

[23] J. Behrens, N. Rakowsky, W. Hiller, D. Handorf, M. Läuter, J. Päpke, K. Dethloff, amatos: Parallel adaptive mesh generator for atmospheric and oceanic simulation, *Ocean Modelling* 10, pp.171–183 (2005).

[24] P. K. Kundu, *Fluid Mechanics*. Academic Press, 638 pp. (1990).