Workshop Proceedings "Current Development in Shallow Water Models on the Sphere "
March, 10-14, 2003, Munich University of Technology, Germany

An adaptive Lagrange-Galerkin shallow-water model on the sphere

M. Läuter, D. Handorf, K. Dethloff
Department Climate System
Alfred Wegener Institute, Foundation for Polar and Marine Research
Postbox 600149, D-14401 Potsdam, Germany
E-mail: mlaeuter@awi-potsdam.de

S. Frickenhaus, N. Rakowsky, W. Hiller
Computing Center
Alfred Wegener Institute, Foundation for Polar and Marine Research
Postbox 120161, D-27515 Bremerhaven, Germany

24th February 2003

Abstract

An adaptive parallel spherical shallow-water model is introduced that uses the Lagrange-Galerkin method (semi-Lagrangian method + finite element method (FEM)).

The scalar formulation of the shallow-water equations, its discretisation and numerical realization are described using suitable techniques for spatial adaptivity, e.g., a stable discretization method, an adaptive grid generator with the space filling curve approach and a flexible solver interface. Computational results show the experimental convergence of the numerical method.

1 Introduction

In this article we want to present a recently developed numerical model based on the shallow-water equations on the sphere with an adaptive grid. This model was developed in the joint project PLASMA (Parallel LArge Scale Model of the Atmosphere). PLASMA is founded in the framework of DEKLIM (German Climate Research Programme) by the Federal Ministry of Education and Research Germany. Project partners are S. Frickenhaus, N. Rakowsky and W. Hiller at the Alfred Wegener Institute in Bremerhaven (responsible for parallelization, code-optimization, solver-implementation), J. Behrens and T. Heinze at the Munich University of Technology (responsible for grid generator and conserving properties) and M. Läuter, D. Handorf, K. Dethloff at the Alfred Wegener Institute in Potsdam (responsible for the physical background and numerical methods).

2 Model Equations

The shallow-water equations describe a horizontal flow within a thin layer of fluid having dynamically varying height and a static underlying orography Φ_0. Starting from the vector formulation in [3] on the unit sphere S and with an evolution time T we obtain, by using the horizontal differential operators defined in the Appendix, the equations

$$\begin{aligned} \partial_t \mathbf{u} + \mathbf{u} \cdot \nabla_{\mathrm{S}} \mathbf{u} + \nabla_{\mathrm{S}} \Phi &= -f \mathbf{n} \times \mathbf{u} - |\mathbf{u}|^2 \mathbf{n}, \\ \partial_t \Phi + \mathbf{u} \cdot \nabla_{\mathrm{S}} (\Phi - \Phi_0) + (\Phi - \Phi_0) \operatorname{div}_{\mathrm{S}} \mathbf{u} &= 0, \\ \mathbf{u} \cdot \mathbf{n} &= 0 \end{aligned} \tag{1}$$

on S. In a rotating Cartesian coordinate system centered in the center of the globe $\mathbf{u} : S \times (0, T) \to \mathbb{R}^3$ denotes the velocity field, $\Phi : S \times (0, T) \to \mathbb{R}$ the geopotential,

$f = 2\,\mathbf{\Omega} \cdot \mathbf{n}$ with $\mathbf{\Omega} = (0, 0, \Omega)^{\mathbf{T}}$ the Coriolis parameter and $\mathbf{n}$ the outward normal vector on S. After applying $\mathrm{rot_S}$ and $\mathrm{div_S}$ to equation (1) and defining $\zeta := \mathrm{rot_S}\,\mathbf{u}$, $\delta := \mathrm{div_S}\,\mathbf{u}$ we obtain the scalar formulation

$$\left.\begin{aligned} \partial_t\zeta + \mathbf{u}\cdot\nabla_{\mathrm{S}}\,\zeta + \zeta\,\delta + f\,\delta &= -\mathbf{u}\cdot\nabla_{\mathrm{S}}\,f, \\ \partial_t\delta + \mathbf{u}\cdot\nabla_{\mathrm{S}}\,\delta + \Delta_{\mathrm{S}}\,\Phi - f\,\zeta &= -(\mathbf{n}\times\mathbf{u})\cdot\nabla_{\mathrm{S}}\,f - J(\mathbf{u}), \\ \partial_t\Phi + \mathbf{u}\cdot\nabla_{\mathrm{S}}(\Phi-\Phi_0) + (\Phi-\Phi_0)\,\delta &= 0 \end{aligned}\right\} \tag{2}$$

with the Helmholtz decomposition

$$\left.\begin{aligned} -\Delta_{\mathrm{S}}\,\psi &= \zeta, \\ \Delta_{\mathrm{S}}\,\chi &= \delta, \\ \mathbf{rot_S}\,\psi + \nabla_{\mathrm{S}}\,\chi &= \mathbf{u} \end{aligned}\right\} \tag{3}$$

on S, where ψ and χ are the streamfunction and the velocity potential. The functional J is defined as

$$J(\mathbf{u}) := (\nabla_{\mathrm{S}}\,u_i)_k\,\delta_{i,l}\,\delta_{j,k}(\nabla_{\mathrm{S}}\,u_j)_l + u_i\,u_j\;\mathrm{div}(\partial_i n_j\,\mathbf{n}).$$

The equations (2) and (3) are the model equations.

3 Numerical Method

The Lagrange-Galerkin method consists in a semi-Lagrangian timestep method and a FEM for the spatial discretization and has been used for spherical shallow-water equations in [7]. To provide grid adaptation with the help of error estimators good stability behavior plays a decisive role.

In the case of the linear advection we know that the semi-Lagrangian method is unconditionally stable, see [8]. Due to the lack of analytical results for the shallow-water equations we assume good stability behavior in the present case, too.

The FEM is very flexibly applicable, see e.g., [6]. In particular, the method is independent of the sizes and arrangement of the grid's elements and one can easily combine elements of different order. These properties make the FEM well suited for the presented adaptive model.

The spatial distribution of grid points in an adaptive model should be controlled by a posteriori error estimators. The used triangulation based on the icosahedral grid is refinable and coarsable independent of the well known pole problem or singularities and thus, it is the appropriate structure for discretization.

3.1 Discretization

The time discretization consists in two main steps. First, we compute the velocity field $\mathbf{u}$ explicitly from vorticity and divergence with the help of the Helmholtz decomposition (3). In doing so it is necessary to solve two Poisson equations (see section 3.3) and to compute derivatives of the potentials. The resulting velocity field is inserted as explicit data in the model equations (2).

In the second step, we apply the semi-Lagrangian method by substituting the total derivatives in the equations (2) by the difference quotient according to the method of characteristics, see [9]. This means, we introduce the trajectory function $\mathbf{X} : S \times (0,T) \times (0,T) \to S$ as a solution of the initial value problem

$$\partial_\tau \mathbf{X}(\mathbf{x},\tau,t) = \mathbf{u}(\mathbf{X}(\mathbf{x},\tau,t),\tau), \quad \mathbf{X}(\mathbf{x},t,t) = \mathbf{x}$$

for every position $\mathbf{x} \in S$ and times $t, \tau \in (0,T)$. For an arbitrary function $\phi : S \times (0,T) \to \mathbb{R}$ this leads to

$$\partial_t\phi(\mathbf{x},t) + \mathbf{u}\cdot\nabla_{\mathrm{S}}\,\phi(\mathbf{x},t) = \partial_\tau\phi(\mathbf{X}(\mathbf{x},\tau,t),\tau)|_{\tau=t}.$$

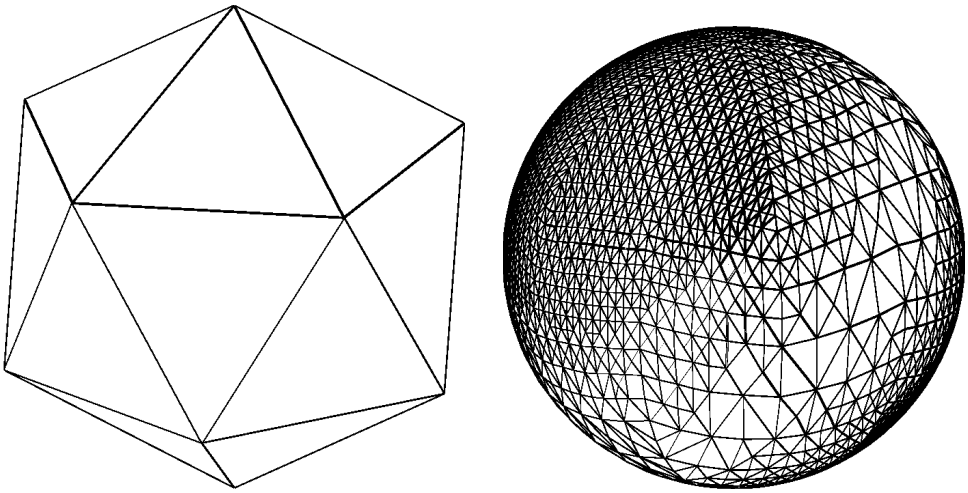

Figure 1: Icosahedral macrotriangulation and adaptive grid according to an a posteriori error estimator

For a timestep Δt we now substitute the right hand side by the difference quotient and obtain

$$\partial_t \phi + \mathbf{u} \cdot \nabla_{\mathrm{S}} \phi(\mathbf{x}, t) \cong \frac{1}{\Delta t} \left(\phi(\mathbf{x}, t) - \phi(\mathbf{X}(\mathbf{x}, t - \Delta t, t), t - \Delta t) \right).$$

In the model, $\mathbf{X}(\mathbf{x}, t - \Delta t, t)$ is approximated with an appropriate Runge-Kutta method. For the model equations (2) this leads to the time discretized version

$$\left.\begin{aligned} \zeta(\mathbf{x}, t) + \Delta t\, \zeta(\mathbf{x}, t)\, \delta(\mathbf{x}, t) + \Delta t\, f\, \delta(\mathbf{x}, t) &= f_1(\mathbf{x}, t), \\ \delta(\mathbf{x}, t) + \Delta t\ \Delta_{\mathrm{S}}\, \Phi(\mathbf{x}, t) - \Delta t\, f\, \zeta(\mathbf{x}, t) &= f_2(\mathbf{x}, t), \\ \Phi(\mathbf{x}, t) + \Delta t\, \Phi(\mathbf{x}, t)\, \delta(\mathbf{x}, t) - \Delta t\, \Phi_0\, \delta(\mathbf{x}, t) &= f_3(\mathbf{x}, t) \end{aligned}\right\} \tag{4}$$

for some functions f_1, f_2, f_3 which contain data from the old timestep.

The remaining spatial derivatives are treated with the FEM leading to a nonlinear system of equations (nonlinearities in the first and last equation of (4)). The nonlinearity is solved by an iteration process that is truncated after a fixed number of iterations (one or two). In the end, we have to solve a linear system of equations (see section 3.3) yielding the new values on the grid for ζ, δ and Φ simultaneously.

3.2 Grid structure

The presented model uses the grid generator `amatos` [1] which was invented by J. Behrens, see [2]. `amatos` works with a Fortran 90 interface, is able to store the model data, adapts the grid structure according to a given refinement criterion, uses hierarchical data structures, supports grid partitioning by the space filling curve (SFC) approach and is available in a shared memory parallel version with OpenMP.

In Fig. 1 the initial icosahedral macrotriangulation and the adapted grid according to an a posteriori error estimator, generated with `amatos`, is to be seen. The construction of a grid partitioning by cutting the SFC into equal chunks of elements is depicted in Fig. 2.

When implementing PLASMA with `amatos`, we had to extend the construction of the SFC through the elements from plane to spherical geometries. At the same

[1] http://www-m8.mathematik.tu-muenchen.de/m3/software/amatos

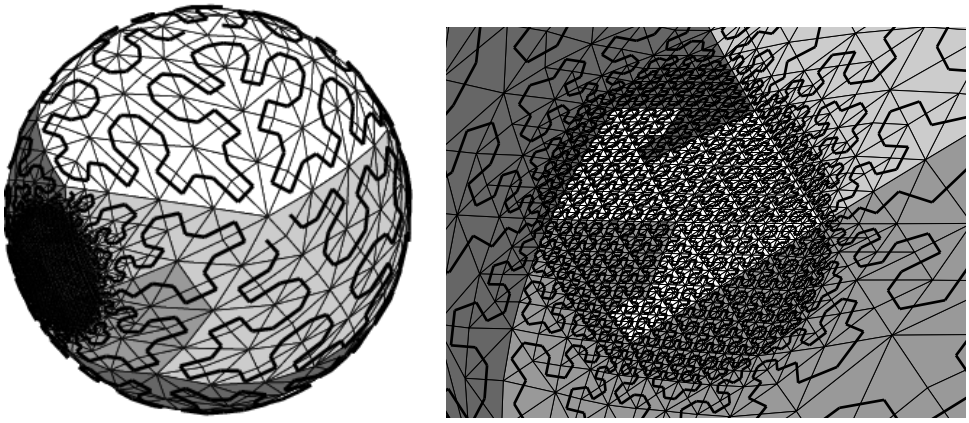

Figure 2: Space filling curve through the elements of a spherical grid with a resulting grid partitioning on 6 processors and a zoom into a refined region

time, we made more information accessible by calculating an SFC through the vertices and edges, too, and by providing all information (e.g., element vertices, nodes coordinates, values at nodes) in arrays sorted along the SFC which also allows to profit from the high data locality on cache level. The impact of the SFC numbering of vertices on the sparsity pattern of a typical stiffness matrix from our model can be studied in Fig. 3 where most entries are gathered close to the diagonal.

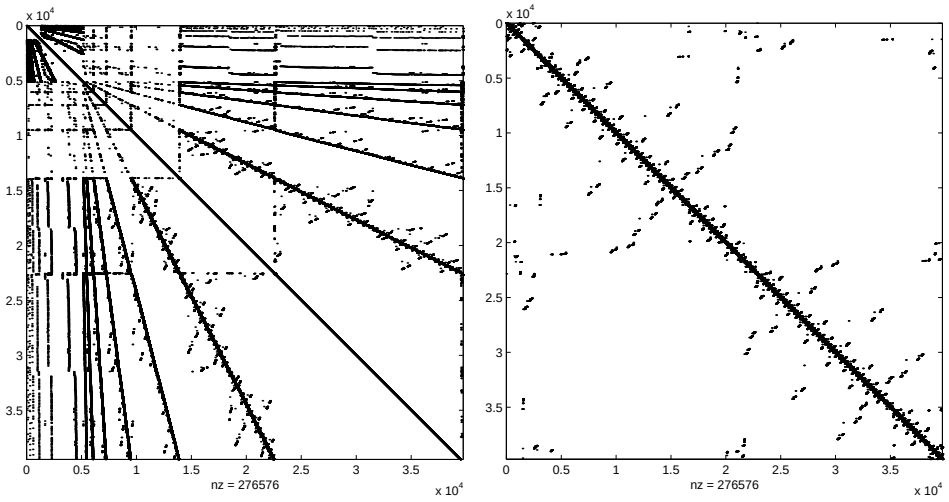

Figure 3: Sparsity pattern of a stiffness matrix for an adaptively refined grid with original numbering (nodes in order of creation) and SFC numbering of nodes

3.3 Linear Solver

The problem of the solution of large linear systems of equations (see section 3.1) requires very efficient computational methods. Especially parallel methods are convenient to obtain reasonable performance. Hence the model uses the SFC approach to realize a good grid partitioning and the parallel solver interface FoSSI (Family of Simplified Solver Interfaces), see [5].

The model's matrices are solved via the FoSSI-interface to PETSC (see [1]), with

the restarted GMRES iterative solver. The restricted additive Schwarz method for parallel preconditioning is applied with ILU(k) or ILUT preconditioners on the subdomains obtained from the SFC. FoSSI offers a highly scalable and flexible solver-setup that allows efficient model simulations on parallel computers. Currently the solver part of the model speeds up for up to 24 CPUs on a SUN-Fire 6800 system with 1050MHz UltraSPARC III CPUs.

4 Computational Results

For a first validation of the model an analytical solution is very convenient, because the numerical error can be computed exactly.

Here we consider the solid body rotation on the unit sphere S with varying axes. The initial condition depending on the angle α for the inclination of the rotation axis is given by

$$\begin{aligned}\mathbf{u}(\mathbf{x}) &= c\,\mathbf{a}(\alpha)\times\mathbf{x} \quad \Leftrightarrow \quad \zeta(\mathbf{x}) = c\,2\,\mathbf{a}(\alpha)\cdot\mathbf{x}, \quad \delta(\mathbf{x}) = 0,\\ \Phi(\mathbf{x}) &= -\frac{1}{2}\left[(c\,\mathbf{a}(\alpha)\cdot\mathbf{x})^2 + 2\,c\,\mathbf{a}(\alpha)\cdot\mathbf{x}\;\boldsymbol{\Omega}\cdot\mathbf{x}\right] + C\end{aligned}$$

with $u_0 = 30\frac{m}{s}$, earth radius R, an arbitrary constant C and

$$c := \frac{u_0}{R}, \quad \mathbf{a}(\alpha) := \begin{pmatrix}-\sin(\alpha)\\ 0\\ \cos(\alpha)\end{pmatrix}, \quad \boldsymbol{\Omega} := \begin{pmatrix}0\\ 0\\ \Omega\end{pmatrix}.$$

With the underlying orography

$$\Phi_0(\mathbf{x}) = \frac{\Omega^2}{2} - \frac{(\boldsymbol{\Omega}\cdot\mathbf{x})^2}{2}$$

we obtain a time periodic quasistationary flow. In Fig. 4 we see the initial conditions for the velocity field $\mathbf{u}$ and the geopotential ϕ.

	number of nodes					number of nodes			
Δt in s	642	1282	2562	5122	Δt in s	642	1282	2562	5122
3600	0.58	0.33	0.21	0.14	3600	0.49	0.32	0.27	0.25
1800	0.60	0.33	0.21	0.14	1800	0.49	0.29	0.23	0.19
900	0.59	0.33	0.21	0.14	900	0.48	0.26	0.19	0.15
450	0.59	0.33	0.21	0.14	450	0.47	0.24	0.18	0.12

Table 1: Relative $L^2(S)$ - error of the geopotential Φ after an iteration time of two days, cases $\alpha = 0$ (left) and $\alpha = \frac{\pi}{4}$ (right)

In table 1 the relative $L^2(S)$ - error is printed for an integration time of two days. We see, that the error decreases for an increasing spatial and time resolution. This means, that we have shown for these two experiments the experimental convergence of the numerical method.

5 Outlook

The project PLASMA focuses on the investigation of internally generated variability of the earth's atmosphere. In order to achieve this goal we will run this model for 30 days up to several months. This requires a further improvement, e.g., of the physical error estimators, of the numerical methods and parameters and of the conservation properties.

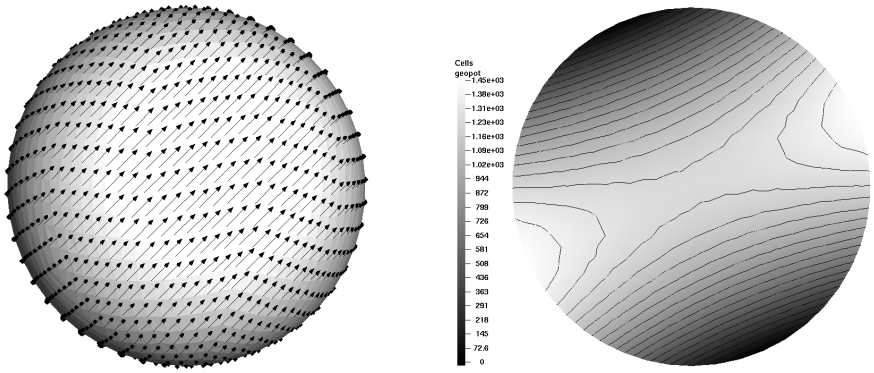

Figure 4: Initial conditions for the velocity field $\mathbf{u}$ and the geopotential ϕ (unit of measurement meter) for the case $\alpha = \frac{\pi}{4}$

Acknowledgments

This work has benefited from the continuous support and encouragement of Jörn Behrens from Munich University of Technology and Eberhard Bänsch from the Weierstrass Institute for Applied Analysis and Stochastics, Berlin.

Appendix

According to [4] we define for arbitrary functions $g : S \to \mathbb{R}$ and $\mathbf{v} : S \to \mathbb{R}^3$ the spherical differential operators

$$\begin{aligned}
\nabla_{\mathrm{S}}\, g &:= \nabla \tilde{g} - (\mathbf{n} \cdot \nabla \tilde{g})\, \mathbf{n}, \\
\mathrm{div}_{\mathrm{S}}\, \mathbf{v} &:= \mathrm{div}\, \tilde{\mathbf{v}} - \mathrm{div}((\tilde{\mathbf{v}} \cdot \mathbf{n})\, \mathbf{n}), \\
\mathrm{rot}_{\mathrm{S}}\, \mathbf{v} &:= \mathbf{n} \cdot \mathbf{rot}\, \tilde{\mathbf{v}}, \\
\mathbf{rot}_{\mathrm{S}}\, g &:= \mathbf{rot}(\tilde{g}\, \mathbf{n}), \\
\Delta_{\mathrm{S}}\, g &:= \mathrm{div}_{\mathrm{S}}\, \nabla_{\mathrm{S}}\, g
\end{aligned}$$

on S. $\tilde{g}$ and $\tilde{\mathbf{v}}$ are smooth continuations of g and $\mathbf{v}$ to a neighborhood of S.

References

[1] S. Balay, W. D. Gropp, et al. Petsc users manual. Technical Report ANL-95/11 - Revision 2.1.3., Argonne National Laboratory, http://www-fp.mcs.anl.gov/petsc, 2002.

[2] J. Behrens. An adaptive semi-Lagrangian advection scheme and its parallelization. *Mon. Wea. Rev.*, 124(10):2386–2395, 1996.

[3] J. Côté. A Lagrange multiplier approach for the metric terms of semi-Lagrangian models on the sphere. *Q. J. R. Meteorol. Soc.*, 114:1347–1352, 1988.

[4] G. Dziuk. Finite elements for the Beltrami operator on arbitrary surfaces. In S. Hildebrandt and R. Leis, editors, *Partial Differential Equations and Calculus of Variations*. Springer, Berlin, 1988.

[5] S. Frickenhaus. *Family of Simplified Solver Interfaces - FoSSI*. Alfred Wegener Institute for Polar and Marine Research, Sientific Computing, http://www.awi-bremerhaven.de/InfoCenter/IT/WorkingGroups/SciComp/SSolversUG, 2002.

[6] C. Großmann and H.-G. Roos. *Numerik partieller Differentialgleichungen*. Teubner, Stuttgart, 1994.

[7] T. Heinze and A. Hense. The shallow water equations on the sphere and their Lagrange-Galerkin-solution. *Meteorol. Atmos. Phys.*, 81:129–137, 2002.

[8] K. W. Morton, A. Priestley, et al. Stability of the Lagrange-Galerkin method with non-exact integration. *Math. Mod. Num. Anal.*, 22:625–653, 1988.

[9] O. Pironneau. *Finite element methods for fluids*. Masson, Paris, 1989.