The BRIDGE project – Basic Research for ICON with Discontinuous Galerkin Extension

44th EWGLAM / 29th SRNWP meeting, Bruxelles

26-29 September 2022

Michael Baldauf, Florian Prill (DWD)

The **BRIDGE** project (**B**asic **R**esearch for **I**CON with **DG E**xtension)

BRIDGE is an *informal* project *at DWD.*

currently: *F. Prill, M. Baldauf*
joining later: *D. Reinert, U. Schättler, S. Borchert*, …

Goals:

- develop a prototype for a Discontinuous Galerkin (DG) implementation of the 3D Euler equations (‚DG-HEVI on the sphere')
- together with a minimal set of physical parameterizations
- using ICON infrastructure (parallelisation, I/O, ...)
- more object-orientation and use of standard software (e.g. YAC coupler, ...)

→ **BRIDGE** is an intermediate step to a full–fledged ICON implementation.

DG development at DWD

COSMO-DG ~2009-2014

- 3D Euler eqns, explicit time integr., structured grid

MetStröm

2D toy model (C++ code) ~2014 – 2020

- How to bring DG on the sphere/ellipsoid (*Baldauf (2020) JCP)*
- HEVI scheme with DG (*Baldauf (2021) JCP)*

BRIDGE 2020 - ~2023

(**B**asic **R**esearch for **I**CON with **DG E**xtension)

- Shallow water on the sphere/ellipsoid (explicit)
- 3D Euler equations (HEVI) + parametr.

ICON-DG ~ 2024 (?) - ~ 2028 (?)

Discontinuous Galerkin (DG) methods in a nutshell (I)

$$\frac{\partial q^{(k)}}{\partial t} + \nabla \cdot \mathbf{f}^{(k)}(q) = S^{(k)}(q), \qquad k = 1, ..., K$$

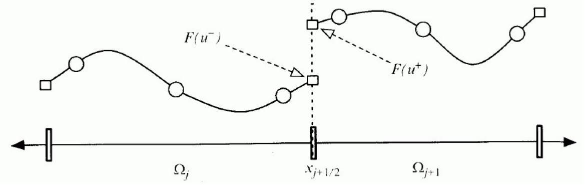
e.g.
Cockburn, Shu (1989) Math. Comput.
Cockburn et al. (1989) JCP
Hesthaven, Warburton (2008)

1.) weak formulation $\int_{\Omega_j} dx\, v(\mathbf{x}) \cdot \ldots$

$$\rightarrow \quad \frac{d}{dt}\int_{\Omega_j} q^{(k)}\, v\, dV + \int_{\partial\Omega_j} f^{(k)num,\perp}\, v\, da - \int_{\Omega_j} \mathbf{f}^{(k)} \cdot \nabla v\, dV = \int_{\Omega_j} S^{(k)}\, v\, dV$$

2.) Finite-element ingredient

$$q^{(k)}(x,t) = \sum_{l=0}^{p} q_{j,l}^{(k)}(t)\; p_l(x - x_j)$$



Galerkin-idea: identify $v \equiv p_l$

Modal base: orthogonal functions e.g. Legendre-Polynomials
Nodal base: interpolation (Lagrange) polynomials

From *Nair et al. (2011)* in
‚Numerical techniques for global atm. models'

Discontinuous Galerkin (DG) methods in a nutshell (II)

Weak formulation

$$\frac{d}{dt}\int_{\Omega_j} q^{(k)}\, v\, dV + \int_{\partial\Omega_j} f^{(k)num,\perp}\, v\, da - \int_{\Omega_j} \mathbf{f}^{(k)} \cdot \nabla v\, dV = \int_{\Omega_j} S^{(k)}\, v\, dV$$

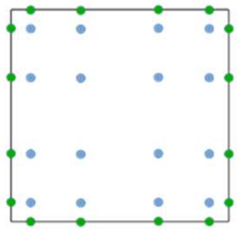
3.) Finite-volume ingredient:

Replace physical flux by a numerical flux in the surface integral
→ couple two neighbouring cells

Often used: simple Lax-Friedrichs flux

$$\mathbf{f}(q) \to f^{num,\perp}(q^+, q^-) = \frac{1}{2}\left(\mathbf{f}(q^+) + \mathbf{f}(q^-)\right) \cdot \mathbf{n} - \frac{\alpha}{2}(q^+ - q^-)$$

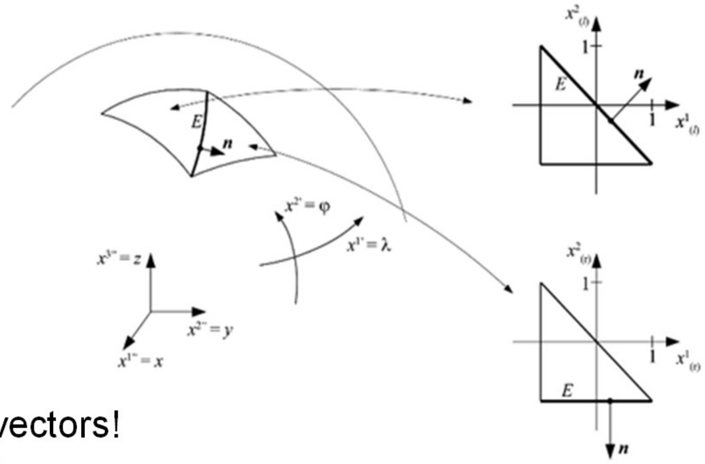
4.) Gaussian quadrature for the volume and surface integrals



→ ODE-system for $q^{(k)}{}_{jl}(t)$

5.) Use a time-integration scheme (Runge-Kutta, …)

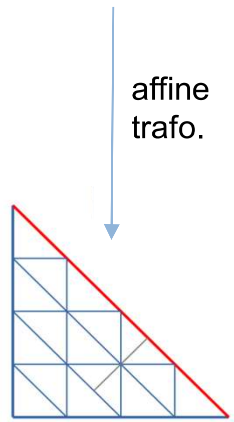
Sequence of coordinates and their transformations in BRIDGE



1. Geographical coordinate
for external data and general I/O purposes

2. Local (gnomonial) coordinate
maps a **spherical** or **ellipsoidal** (or just **flat**) triangle
to a unit triangle.
Use covariant form of eqns. (Ricci tensor calculus)
→ avoids any singularity both in coordinates and in base vectors!
~ concept of a differentiable manifold (*Baldauf (2020) JCP*)

3. Unit coordinate
affine trafo., compatible to standard FE codes



4. Terrain-following coordinate
treat orography, use strong conservation form (*Baldauf (2021) JCP*)

5. Submapping coordinate
affine trafo., reduce number of flux transf. at edges

Extension to the 3D Euler equations on the sphere together with terrain-following coordinates

Additional metric terms of terrain-following coordinates can destroy numerical local conservation → use *strong conservation form* of the equations, i.e. use both base vectors for a smooth (e.g. spherical) coordinate system K' and for the terrain-following system K (*Wedi, Smolarkiewicz (2003), Baldauf (2021)*)

example: strong cons. form of the momentum equation:

$$\frac{\partial}{\partial t}\sqrt{g}\,M^{i'} + \frac{\partial}{\partial x^k}\sqrt{g}\,T^{i'k} = \sqrt{g}\,(S^{i'}_{(M)} - \Gamma^{i'}_{k'l'}T^{l'k'})$$

now: additional metric terms only from the smooth system K'

momentum flux for Euler eqns.

$$T^{ik} = \frac{1}{\rho}M^i M^k + \tilde{p} g^{ik}$$

for diffusion (D^{ik} = deformation tensor), add

$$T^{ik}_{diff} = -\rho K_a 2 D^{ik} - \rho K_b g^{ik} \nabla_l v^l$$

Additionally: Continuity eq. (for ρ) and energy equation (for $\rho\theta$)

A general remark about this talk:

at the last SRNWP/EWGLAM-meeting 2021, almost all results shown still have been generated by the **toy model** (C++ code, only 2D, …)

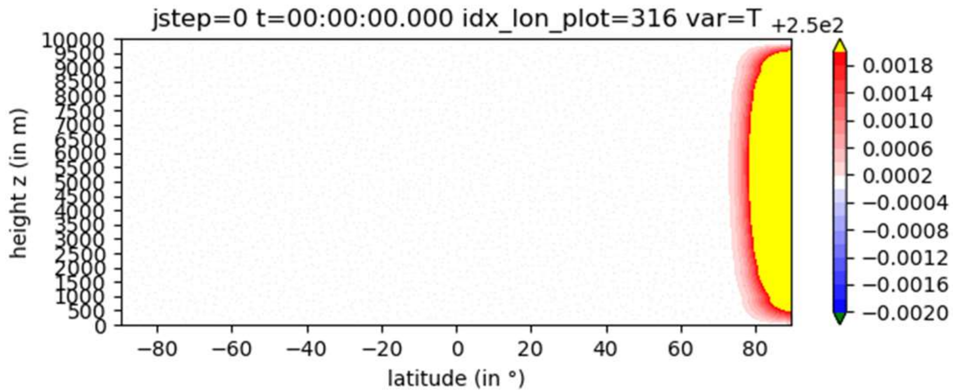
Now, all results are generated by the **BRIDGE** code …

3D Euler equations:
quasi-linear expansion of gravity and sound waves in a spherical shell

Baldauf, Reinert, Zängl (2014) QJRMS derive a linearized analytic solution for

- test scenario (A): $f = 0$
- test scenario (B): $f = 10 * f_{\text{geo}}(45°)$ → $f / N \sim 0.05$

DG 4th order scheme with explicit time integration (4th order Runge-Kutta)

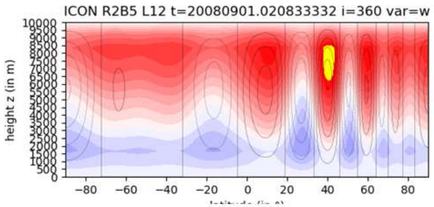


‚small planet':
$R = R_{\text{earth}} / 50$

A very first (!), almost (!) fair comparison between ICON and BRIDGE

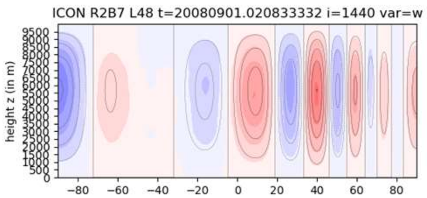
- test case *Baldauf, Reinert, Zängl (2014) QJRMS*: expansion of sound- and gravity waves in a spherical shell (uses shallow atmosph. approx., $U_0=0, f=0$).
- $R= r_{\mathrm{earth}}$ / 100 → almost isotropic grid cells
  → HEVI- (ICON) and purely explicit time integration (BRIDGE) are comparable!
- all simul. on our RCL (ICON: 32, 64, 384 proc. (thanks to D. Reinert!), BRIDGE: 16 proc.)

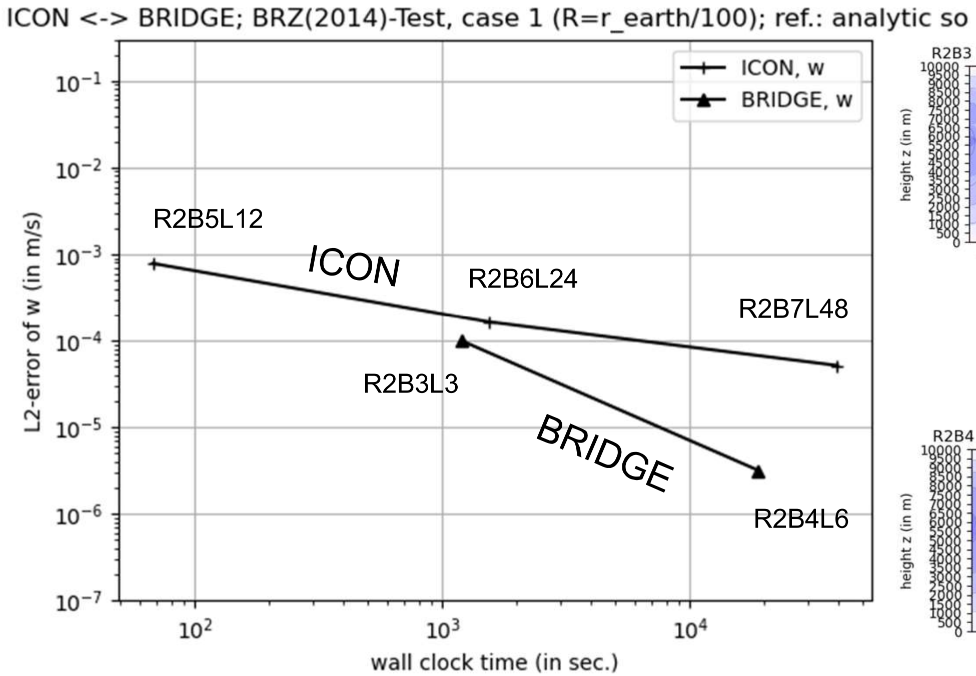
R2B5L12



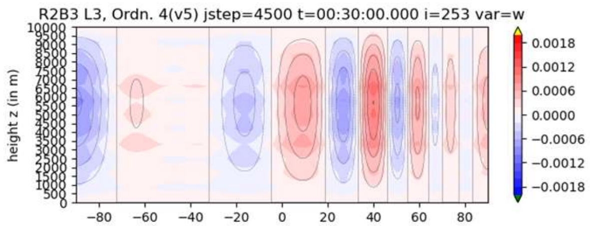
ICON

R2B7L48



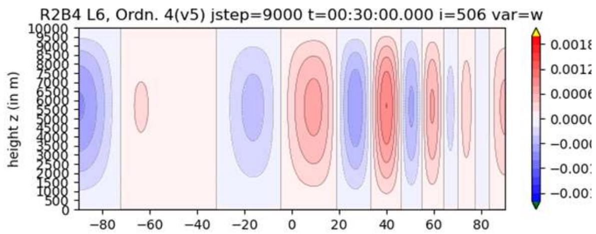


R2B3L3



BRIDGE

R2B4L6

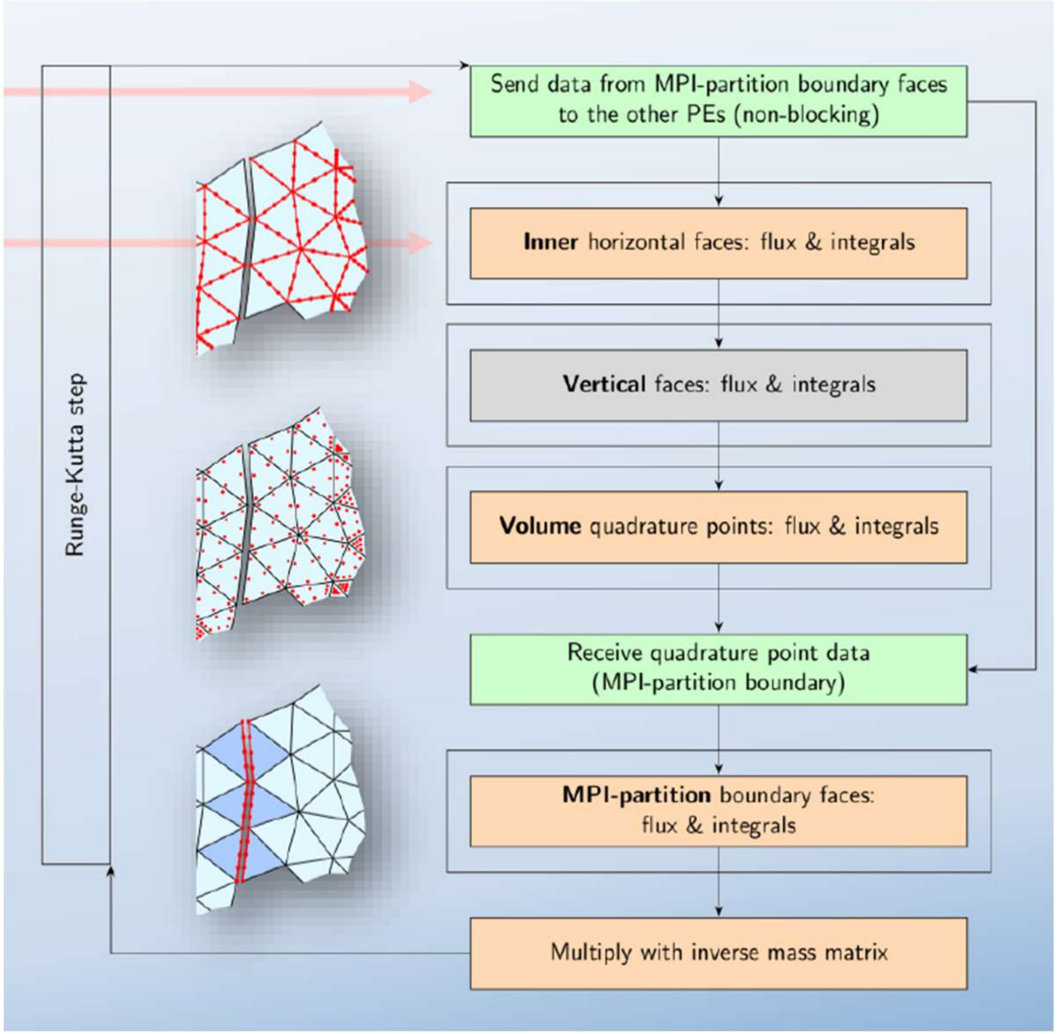


Parallelization in BRIDGE

Goals:

- Minimum number of MPI synchronizations
- Do as much as possible calculations between a send and a receive

DG allows to achieve these goals!



Physical diffusion in a DG scheme

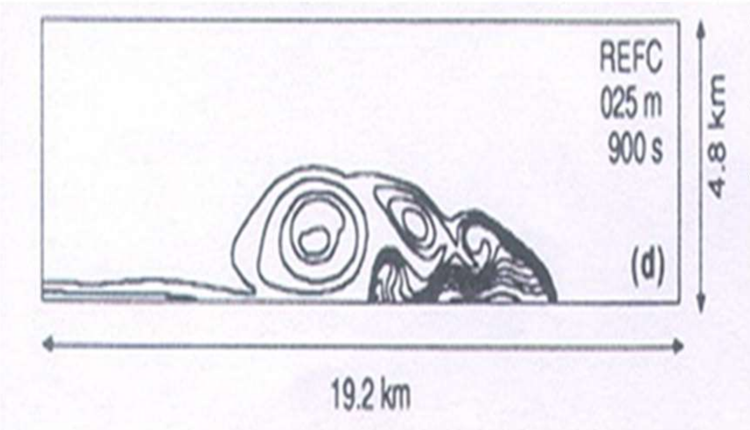
It is a basic result of FE schemes that the 2nd order diffusion operator requires *continuous base functions* (through the whole domain!).
This seems to be in contradiction to the DG method…
→ several approaches exist in the literature

Bassi, Rebay (1997) [BR1]:
determine (spatial) derivative variables $q_{,i} = \mathbf{e}_\mathrm{i} \cdot \mathrm{grad}\ q = \mathrm{div}\ (\mathbf{e}_\mathrm{i}\ q)$
of the prognostic variables q and treat them by the ‚DG procedure‘;
now numerical fluxes don‘t need numerical diffusion
(for inclusion of metric properties see *Baldauf (2021) JCP*).

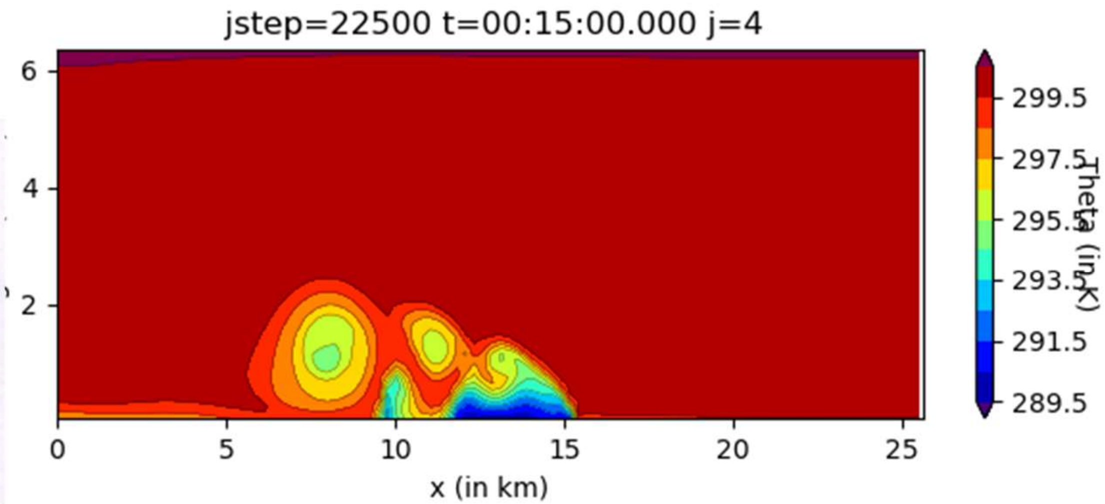
This needs:
- implementation of surface- and volume integrals of the above mentioned ‚pseudo-fluxes‘
- additional source term integrals,
- an externsion of the FE-data structures.
- Implementation of test cases described in *Baldauf, Brdar (2016) QJRMS*

Falling cold bubble in a viscous medium
test setup: Straka et al. (1993)

Reference solution
for θ from
Straka et al. (1993)
($\Delta x=\Delta z=25m$)



BRIDGE, 4th order, RK4, $\Delta x=\Delta z=400m$



Horizontally explicit - vertically implicit (HEVI)-scheme with DG

Motivation: get rid of the **strong time step restriction** by vertical sound wave expansion in **flat grid cells** (in particular near the ground)

$$\frac{\partial q^{(s)}}{\partial t} + \nabla \cdot \mathbf{f}^{(s)}_{slow} + \nabla \cdot \mathbf{f}^{(s)}_{fast} = S^{(s)}_{slow} + S^{(s)}_{fast}$$

explicit — implicit — explicit — implicit

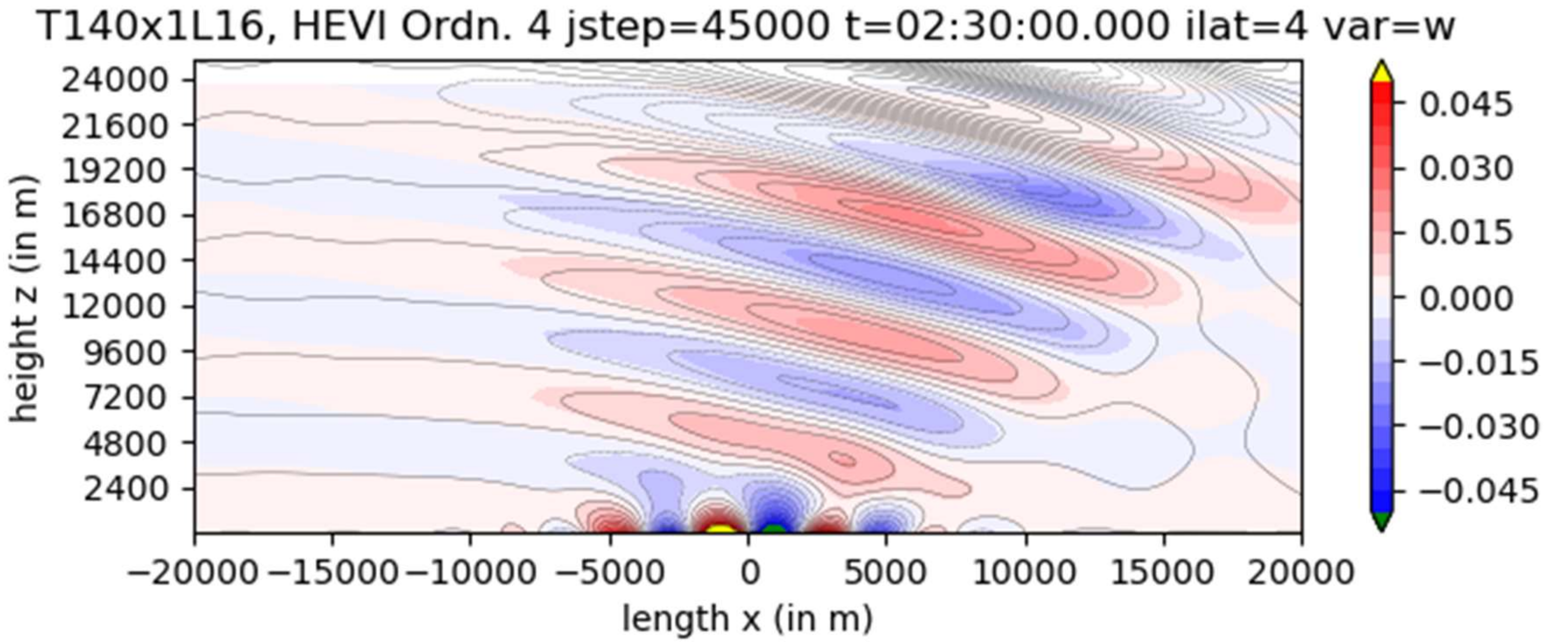
$$\mathbf{f}^{(s)}_{fast} = f^{(s)}_{z,fast}\mathbf{e}_z$$

$$f^{(s)}_{z,fast} = \sum_{s'} H^{ss'} q^{(s')}$$

- The **implicit** part leads to several **block-tridiagonal** matrices
  → here a direct solver is used (expensive!)
- Use of **IMEX-Runge-Kutta** (SDIRK) schemes: SSP3(3,3,2), SSP3(4,3,3) (Pareschi, Russo (2005) JSC)

Baldauf, M. (2021): *A horizontally explicit, vertically implicit (HEVI) discontinuous Galerkin scheme for the 2-dim. Euler and Navier-Stokes equations using terrain-following coordinates*, J. Comp. Phys. 446

Flow over mountains
setup from *Schär et al. (2002)*, N=0.01 1/s, …, with vertical grid stretching
HEVI-solver, 4th order, SSP3-4-3-3 RK scheme, dx=1km, dt=0.2s



Colors: BRIDGE   (1 proc. on rcl, t_wall=21.5h)
Contours: analytic solution (*Baldauf (2008) COSMO-NL*)

Current state of the efficiency of the HEVI solver

In a full **3D simulation** with **DG 4th order**, currently 95% of computation time is spent in the HEVI solver!

- Generation of coefficient matrix elements (26%)
- LU decomposition (22%)

called every 50 time steps

- Solve LSE (23%)
- Calculate implicit tendency (24%)

called every RK substep

Of course, these numbers must be heavily reduced!

There exist ideas in the literature using the so called ‚Schur complement‘ or ‚static condensation‘, which must be tested…
We currently are in contact with the University of Cologne, who have experience with such methods.

The block-tridiagonal coefficient matrices contain ~25% non-zeros (non-collocated) or ~6 % (collocated) → i.e. they are not (very) sparsely occupied

Milestones of the BRIDGE-project (status Sept. 2022)

0.	first version of a **FE/DG framework** available, **MPI** parallelized	Q2/2021 ✓
1.	**shallow water** equations on the sphere, **explicit** time integration (RK)	Q3/2021 ✓
2.	**3D Euler** equations on the sphere, explicit time integration (RK)	Q1/2022 ✓
2.b	with 3D diffusion (+ a simple turbulence scheme)	Q2/2022 ✓/🛠
2.c	grid refinement works	Q2/2022 ✓
3.	**Euler** equations, **HEVI** time integration (IMEX-RK)	Q3/2022 (✓)
3.b	optimization of the vertically implicit solver (Schur compl.,...)	Q4/2022 🛠
3.c	with 3D diffusion (+ a simple turbulence scheme), HEVI	Q1/2023
4.	cloud **microphysics** (Kessler) + **tracer advection** (positive definit) + explicit sedimentation scheme	Q4/2022
4.b	cloud microphysics (cloud ice, Graupel) + vertically implicit sedimentation scheme	Q1/2023
5.	**limited area** version available	Q1/2023
5.b	**vectorized** version available	Q2/2023
6.	coupling of a full fledged **turbulence + BL** scheme	Q2/2023

During all these stages, additional optimization will take place …

Outlook: building blocks for turbulence parameterizations (I)

Keep in mind: we want to *conserve prognostic variables*,
therefore, the turbulence parameterization shouldn‘t deliver tendencies, but:

- fluxes $\boldsymbol{f}^{(s)}$ (for explicit time integration)
- or a coefficient matrix of the form $H^{ss'} \equiv \delta \boldsymbol{f}^{(s)} / \delta q^{(s')}$ (for implicit time integration)
- or a combination of both

note: don‘t bother with the time-integration itself; this takes place in a (hopefully) stable IMEX-RK scheme.

To calculate these fluxes at a certain (quadrature) point $\boldsymbol{r}$,
the turbulence model gets derivative variables at $\boldsymbol{r}$ from the dyn. core
(‚x-, y-, z-‘-derivatives of the prognostic variables ρ, $\boldsymbol{M}$, $\vartheta = \rho\theta$, $q^{(s)}$).
Use the chain rule of differentiation to derive arbitrary other spatial derivatives of a variable $\varphi = f(\rho, \boldsymbol{M}, \vartheta, q^{(s)})$.
In this ideal world, a turbulence modeler wouldn‘t need any grid information!

Outlook: building blocks for turbulence parameterizations (II)

Problems to solve:

- In $\varphi = f(\rho, \boldsymbol{M}, \vartheta, q^{(s)})$, the function f may contain non-differentiable Heaviside-functions (i.e. ‚if-conditions‘) (?). Replace these by ‚relaxation expressions‘?
- How to treat non-local closures? (length scales, smoothing operations, spatial integrations, …)

Planned steps from the ‚DG developers‘ side for the turbulence modelers:

- Deliver a Smagorinsky model
- Implement a prognostic TKE equation as an example
- Deliver diffusion coefficients from this TKE

(Folien von ICCARUS 2022 WG2 meeting)

DG – Pros …

Deutscher Wetterdienst
Wetter und Klima aus einer Hand

- **local conservation** of every prognostic variable ← ‚FV heritage'
- any **order of approximation (convergence)** possible ← ‚FE heritage'
- flexible application on **unstructured grids** (also dynamic adaptation is possible, h-/p-adaptivity)
- very good **scalability** on massively-parallel computers (compact data transfer and no extensive halos)
- **separation** between (analytical) equations and numerical implementation
- **boundary conditions** are easily prescribed (fluxes or values in weak form) → coupling with other subcomponents (ocean model, …) should be easy
- higher accuracy helps to **avoid several awkward approaches** of standard 2nd order schemes: staggered grids (on triangles/hexagons, vertically heavily stretched), numerical hydrostatic balancing, grid imprints by pentagon points or along cubed sphere lines, …
- **unified numerical treatment** of all flux terms and source terms
- **explicit** schemes are relatively easy to build and are quite well understood

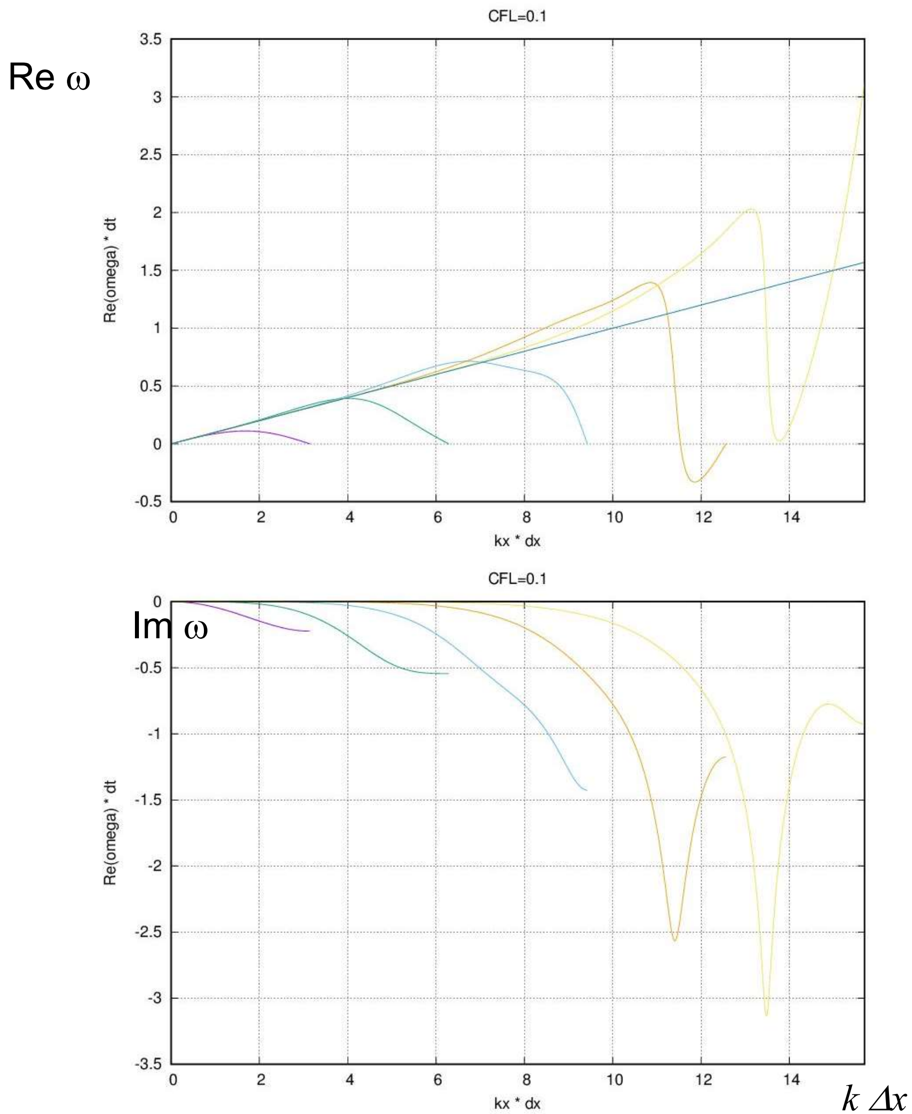
DG – … and Cons

- **high computational costs** due to
  - (apparently) **small Courant numbers** → small time steps
  - higher number of **degrees of freedom**
    - variables ‚live' both on interior *and* on edge quadrature points
    - this holds additionally for **parabolic problems (diffusion)**
  - HEVI approach leads to **block tridiagonal matrices** with larger blocks

→ All these expenses must be outperformed by:
higher convergence order, better computational intensity, and better parallelization!

- **well-balancing** (hydrostatic, perhaps also geostrophic?) in Euler equations is an issue → can be solved!
- basically ‚only' an **A-grid-method** →however, the ‚spurious pressure mode' is very selectively damped!

Dispersionrelation der
1D-Advektionsgleichung
mit DG n-ter Ordnung (n=1..5)



Beitrag von Michael B.

Shallow-water equations in covariant form, i.e. only tensors occur
→ equations are valid *on any 2D manifold* (at least from a mathematical viewpoint)

$$\frac{\partial H}{\partial t} + \nabla_j M^j = 0,$$

$$\frac{\partial M^i}{\partial t} + \nabla_j T^{ij} = S^i, \quad i, j = 1, 2,$$

momentum flux tensor:

$$T^{ij} = \frac{M^i M^j}{H} + \frac{1}{2} g_{grav} H^2 g^{ij},$$

source vector of momentum:

$$S^i = -g_{grav} H g^{ij} \nabla_j h_B + f_c g^{ij} E_{jl} M^l,$$

E_{jl} : 2nd rank Levi-Civita pseudo tensor,
f_c : Coriolis parameter (a pseudo scalar field)

express covariant derviative ∇_j
by partial derivative and Christoffel symbols
→ accessible to a numerical implementation:

$$\frac{\partial \sqrt{g} H}{\partial t} + \frac{\partial}{\partial x^j} \sqrt{g} M^j = 0,$$

$$\frac{\partial \sqrt{g} M^i}{\partial t} + \frac{\partial}{\partial x^j} \sqrt{g}\, T^{ij} + \sqrt{g}\, \Gamma^i_{jk} T^{kj} = \sqrt{g} S^i.$$

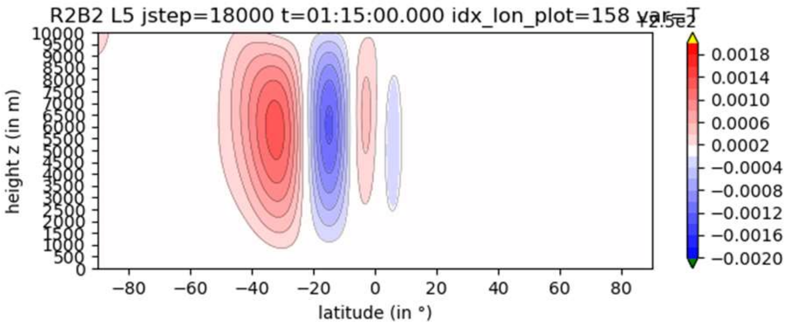
Baldauf, M. (2020): *Discontinuous Galerkin solver for the shallow-water equations in covariant form on the sphere and the ellipsoid,* J. Comp. Phys. 410

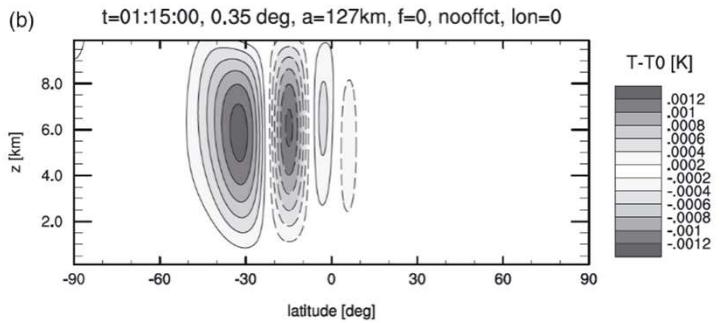
3D Euler equations:
quasi-linear expansion of gravity and sound waves in a spherical shell

Black lines : analytic solution (BRZ2014)
Color shading: BRIDGE / grey shading: ICON

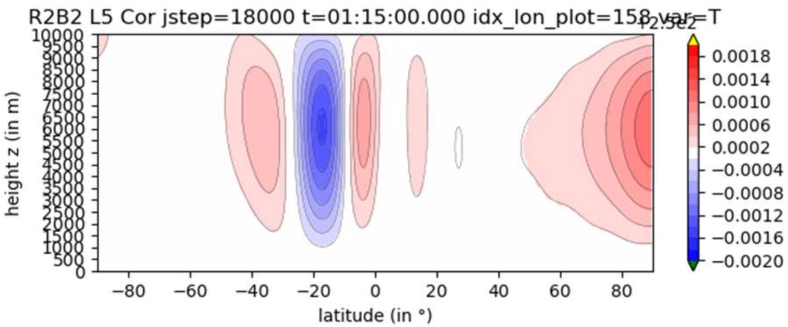
BRIDGE, 4th order, R2B2 L5 | ICON R2B6 L40

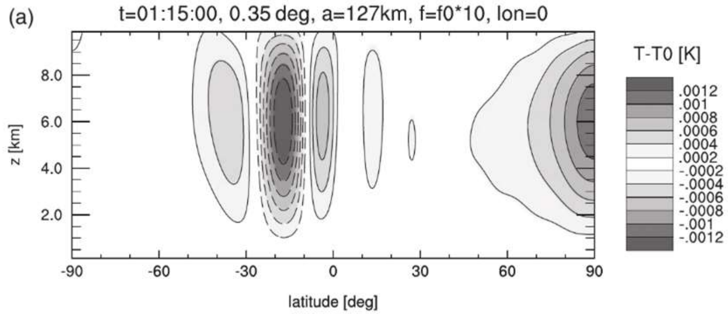
(A):
f=0





(B):
f=10*f(45°)
→
$f\,/\,N$ ~ 0.05





(from *Baldauf et al. (2014) QJRMS*)

Step 2:

extension for the Euler equations in terrain-following coordinates and a HEVI time integration

Dry Euler equations:

$$\frac{\partial \rho}{\partial t} + \nabla_j M^j = 0, \qquad M^j = \rho v^j$$

$$\frac{\partial M^i}{\partial t} + \nabla_j \left(\frac{M^i M^j}{\rho} + p g^{ij} \right) = -g k^i \rho - 2 \epsilon^i{}_{jk} \Omega^j v^k,$$

$$\frac{\partial \vartheta}{\partial t} + \nabla_j \left(\frac{M^j}{\rho} \vartheta \right) = 0, \qquad \vartheta = \rho \Theta,$$

$$p = p_{ref} \left(\frac{\vartheta R_d}{p_{ref}} \right)^{c_p / c_v}$$

Properties of the dynamical core of ICON

- uses non-hydrostatic, compressible Euler eqns.
- exactly mass- and tracer mass-conserving.
- It is a true 2nd order scheme (as long as parameterizations are switched off).
- stable in very steep mountainous regions.
- useable both for global and regional applications.
- computationally very efficient and scales well on current parallel comp.

(Zängl et al. (2015) QJRMS, Zängl (2012) MWR)

Some numerical details:

- staggering: horizontally: triangle C-grid, vertically: Lorenz-grid
- mixed finite-volume / finite-difference
- predictor-corrector time-integration
- several damping mechanisms are used (divergence damping (2D and quasi-3D), off-centering in the vertically implicit solver, artificial horizontal diffusion…)

IMEX-Runge-Kutta

- general stability function for the Dahlquist problem is known
- general order conditions are known
- described by double Butcher tableaus
  e.g. SSP3(3,3,2) by *Pareschi, Russo (2005) JSC:*

$$
\begin{array}{c|ccc}
0 & 0 & & \\
1 & 1 & 0 & \\
1/2 & 1/4 & 1/4 & 0 \\
\hline
 & 1/6 & 1/6 & 2/3
\end{array}
\qquad\qquad
\begin{array}{c|ccc}
a & a & & \\
1\text{-}a & 1\text{-}2a & a & \\
1/2 & 1/2\text{-}a & 0 & a \\
\hline
 & 1/6 & 1/6 & 2/3
\end{array}
$$

$$a = 1 - 1/\sqrt{2}$$

- practically SDIRK schemes are preferred

Lock, Wood, Weller (2014) QJRMS
Pareschi, Russo (2005) JSC: SSP3(3,3,2), SSP3(4,3,3)
Giraldo et al. (2012) Siam JSC: ARK2(2,3,2)
Kang, Giraldo, Bui-Thanh (2020) JCP: IMEX-RK in hybridiz. DG

IMEX-Runge-Kutta

RK stages:

$$\boldsymbol{q}^{(i)} = \boldsymbol{q}^n + \Delta t \sum_{j=1}^{i-1} \beta_{ij}^{(ex)} \boldsymbol{S}^{(j)} + \Delta t \sum_{j=1}^{i} \beta_{ij}^{(im)} \boldsymbol{F}^{(j)}$$

+ a final step that combines all tendencies $\boldsymbol{S}^{(j)}$ and $\boldsymbol{F}^{(j)}$ → $\boldsymbol{q}^{n+1}$

- coefficients β are described by double Butcher tableaus
  e.g. SSP3(3,3,2) by *Pareschi, Russo (2005) JSC:*

$$\begin{array}{c|ccc} 0 & 0 & & \\ 1 & 1 & 0 & \\ 1/2 & 1/4 & 1/4 & 0 \\ \hline & 1/6 & 1/6 & 2/3 \end{array} \qquad \begin{array}{c|ccc} a & a & & \\ 1\text{-}a & 1\text{-}2a & a & \\ 1/2 & 1/2\text{-}a & 0 & a \\ \hline & 1/6 & 1/6 & 2/3 \end{array} \qquad a = 1 - 1/\sqrt{2}$$

- general stability function for the Dahlquist problem is known
- general order conditions are known
- practically SDIRK schemes are preferred

Lock, Wood, Weller (2014) QJRMS
Pareschi, Russo (2005) JSC: SSP3(3,3,2), SSP3(4,3,3)
Giraldo et al. (2012) Siam JSC: ARK2(2,3,2)
Kang, Giraldo, Bui-Thanh (2020) JCP: IMEX-RK in hybridiz. DG

Flow over mountain with the HEVI-solver

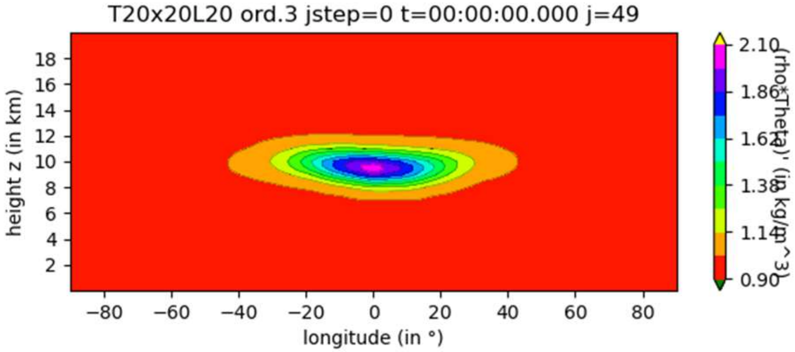
Setup : Schär et al. (2002)

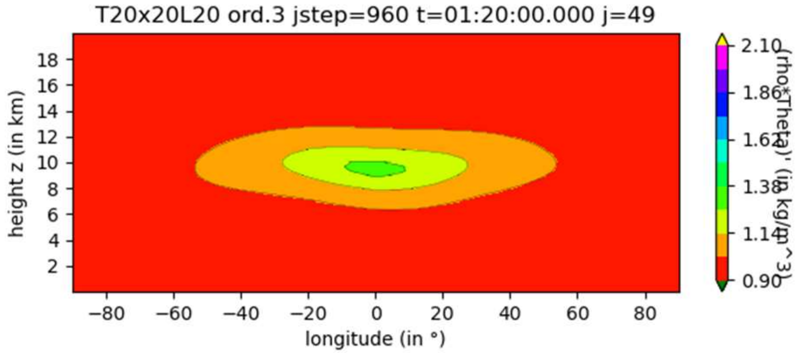
Orography: $$h(x) = h_0 \cdot e^{-x^2/b^2} \cdot \cos^2 \pi \frac{x}{\lambda}$$

h_0=10m, b=5km, λ=4km → Fr_h=100, Fr_a=0.1 … 0.5
u_0=10m/s, N=0.01 1/s, $T(z=0)$=288K

compare with analytic linear solution: *Baldauf, 2008, COSMO-NL*
(uses only a few further approximations, e.g. it is a fully compressible solution)

Scalar diffusion over flat plane, with orography (analyt.sol.: black contours):



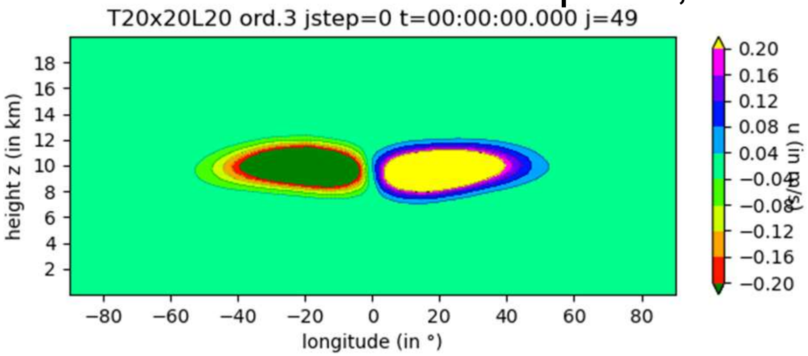


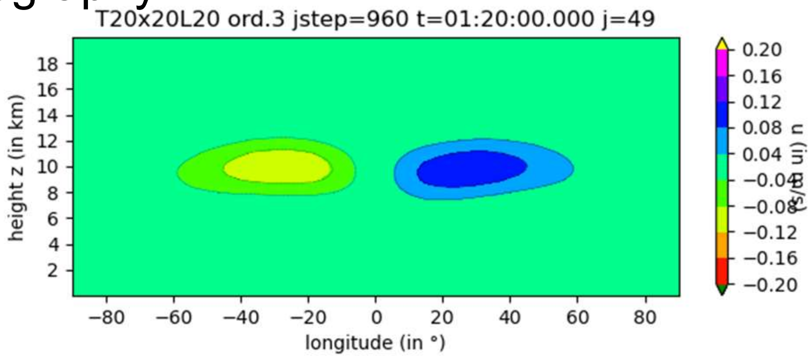
Feld 1: Min=0.999978956050552, Max=2.0422468996731746
Feld 2: Min=0.999978956050552, Max=2.0422468996731746

Feld 1: Min=0.9985417513694614, Max=1.2841676764579049
Feld 2: Min=0.9999999899444169, Max=1.2834290212464434

correct convergence behaviour ☺

Vectorial diffusion over flat plane, with orography





Feld 1: Min=-0.6438439534887065, Max=0.6162612019564722
Feld 2: Min=-0.6438439534887059, Max=0.6162612019564725

Feld 1: Min=-0.11587963802688567, Max=0.11556142718881962
Feld 2: Min=-0.11657519071791987, Max=0.11619112763452501

correct convergence behaviour ☺

Flow over mountains with steep slopes and vertical grid stretching

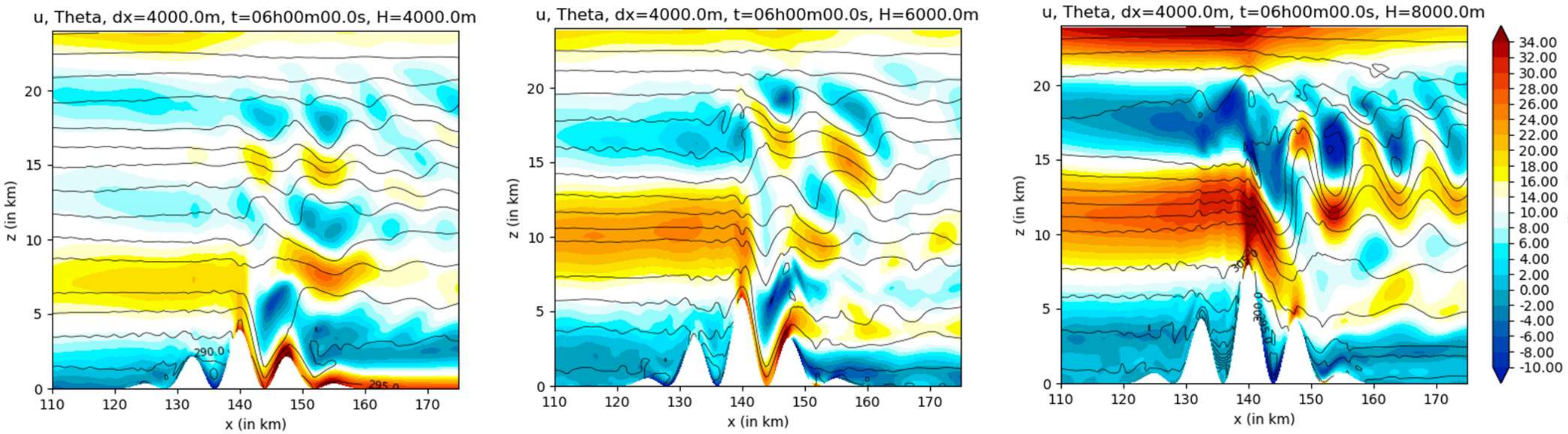
Schaer et al. (2002) MWR, test case 5b: U_0=10m/s, N=0.01 1/s, but a=10km

Results from 2D toy model

H_{oro}= 4000m, α_{max} = 58°

H_{oro}= 6000m, α_{max} = 67°

H_{oro}= 8000m, α_{max} = 72°



Δx=4 km; vertical grid stretching: Δz_{min}~46m, Δz_{max}~736m, $z_{lowest\ QP}$~10.3m

HEVI-DG simulation (4th order) remains stable even for steeper slopes!

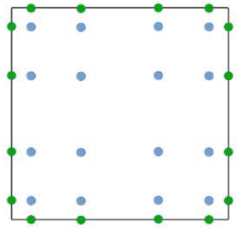
to avoid instability by strong gravity wave breaking, vertically implicit ‚3D' Smagorinsky diffusion was used

Additionally done

- Method for a consistent use of real orography data (→ EXTPAR)
- Treatment of diffusion in a HEVI-DG scheme with terrain-following coordinates (local DG by the *Bassi, Rebay (1997) JCP* approach)
- Efficiency improvement of the implicit solver
  (perform expensive LU-decomposition only after several dozen time steps)
- Formulation of boundary conditions for higher order schemes

Baldauf, M. (2021): *A horizontally explicit, vertically implicit (HEVI) discontinuous Galerkin scheme for the 2-dim. Euler and Navier-Stokes equations using terrain-following coordinates*, J. Comp. Phys. 446

What makes DG methods expensive (summary)



- Higher number of degrees of freedom (DOF):
  - quadrature points inside the cell and additionally on the edges
    (some nodal DG methods avoid additional edge points for the price of accuracy)
  - Diffusion (or generally higher order derivatives) by Bassi, Rebay- or local DG-procedure needs treatment of derivative variables analogous to prognostic variables → again increases number of DOFs.
- HEVI scheme needs solution of band diagonal matrices with relatively many off-diagonal bands: currently takes about 60% of the whole calculation
- Small Δt
  - CFL numbers are smaller than one would expect; in particular for IMEX-RK
    ICON has CFL=0.6; for 4th order DG one would expect CFL~0.15, but SSP3(4,3,3) → 0.079
  - Δt is not only determined by c_{snd} but by $c_{\mathrm{snd}} + |v_{\mathrm{max}}|$

→ All these expenses must be outperformed by:
higher convergence order, better computational intensity, and better parallelization!

DG and Parameterisations

General principle: spatial transport (advection, sedimentation, diffusion, …) must be treated in the DG-scheme!
Otherwise we loose local conservation.

Box-models (e.g. **cloud physics**, **chemistry/aerosol-packages**):
are evaluated and deliver tendencies in every quadrature point
→ at the first place no adaptations necessary!
Nevertheless, the ‚classical' physics/dynamics-coupling questions remain:
overall time integration scheme? how to achieve positive definiteness?

Turbulence:
Remark: diffusion needs special treatment in DG (local DG, compact DG, …)
Advantage of local DG: derivatives of fields are directly available for turbulence modeling!

Analogous: convection param., gravity wave param., …

DG and physics perturbations in ensembles

Some recommendations …
… to keep conservation properties of the DG scheme:

- in the **transport terms**, only (physical) fluxes should be perturbed.
- in the **source terms**: e.g. moisture var., perturb in a way that $\rho_{dry} + \rho_v + \rho_r + ... + \rho_g$ is unchanged (while keeping positive def.)

DG and data assimilation

At least adaptations in the forward operators necessary:

- by the modified output grid; better say: to the position of the I/O-grid points (these are probably the quadrature points in the triangle grid)
- different prognostic variables (conserved var.)