A Dynamically Adaptive Wavelet-based Method for Geophysical Flows on the Sphere

Nicholas Kevlahan

Department of Mathematics & Statistics

McMaster University

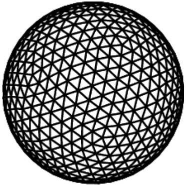
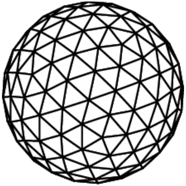
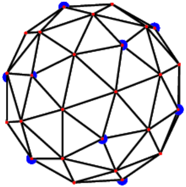
PDEs on the Sphere 2014

Collaborators

- Matthias Aechtner
  PhD student, Computational Science and Engineering
  McMaster University
- Thomas Dubos
  Laboratoire de Météorologie Dynamique
  École Polytechnique, France

Method

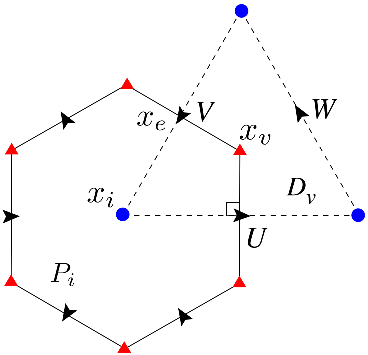
Discrete wavelet transform on the sphere

vertex values	$\Rightarrow$	vertex values	$\Rightarrow$	vertex values	***restriction***
$\Downarrow$		$\Downarrow$			***high-pass filter***
interpolation errors		interpolation errors			***wavelets***

interpolation errors		interpolation errors			***wavelets***
$\Downarrow$		$\Downarrow$			***reconstruction***
vertex values	$\Leftarrow$	vertex values	$\Leftarrow$	vertex values	***prolongation***

TRiSK scheme *(Thuburn et al. 2010)*



Staggered dual grids for pressure and vorticity
(Velocity at cell edges)

Discrete shallow water equations

$$\frac{\partial h_i}{\partial t} = -[\text{div}(F_e)]_i$$
$$\frac{\partial \mathbf{u}_e}{\partial t} = F_e^{\perp}\hat{q}_e - [\text{grad}(B_i)]_e$$

- $F_e = \hat{h}_e u_e$ is thickness flux
- $F_e^{\perp}$ is perpendicular to F_e

Scale commutation properties of differential operators

$$
\begin{tikzcd}
B_i^0,\ F_e^0,\ u_e^0 \arrow[r, "\mathrm{grad}^0,\ \mathrm{div}^0,\ \mathrm{curl}^0"] & \mathrm{grad}\ B_i^0,\ \mathrm{div}\ F_e^0,\ \mathrm{curl}\ u_e^0 \\
B_i^1,\ F_e^1,\ u_e^1 \arrow[u, "R"'] \arrow[r, "\mathrm{grad}^1,\ \mathrm{div}^1,\ \mathrm{curl}^1"] & \mathrm{grad}\ B_i^1,\ \mathrm{div}\ F_e^1,\ \mathrm{curl}\ u_e^1 \arrow[u, "R"']
\end{tikzcd}
$$

Commutation diagram

Scale commutation properties of differential operators

Commutation relations

$$
\begin{aligned}
R_h^j \circ \mathsf{div}^{j+1} &= \mathsf{div}^j \circ R_F^j && \textit{conserve mass} \\
\mathsf{curl}^j \circ R_{\mathbf{u}}^j &= R_\zeta^j \circ \mathsf{curl}^{j+1} && \textit{conserve circulation} \\
\mathsf{grad}^j \circ R_B^j &= R_{\mathbf{u}}^j \circ \mathsf{grad}^{j+1} && \textit{no spurious vorticity}
\end{aligned}
$$

Volume penalization of shallow water equations

Variable porosity

$$\phi(x) = \alpha + (1 - \chi(x))(1 - \alpha), \quad \alpha \ll 1$$

mask $\chi = 1$ in solid and $\chi = 0$ in fluid.

Volume penalization of shallow water equations

Euler–Poincaré theory

Applying Hamilton's principle of least action to
$\mathcal{L} = \int \frac{1}{2} h(|\mathbf{u}|^2 - gh)\phi \, \mathrm{d}x \, \mathrm{d}y \, \mathrm{d}t$ gives

$$\begin{aligned} \frac{\partial}{\partial t}\tilde{h} + \mathsf{div}\tilde{F} &= 0 \\ \frac{\partial}{\partial t}\tilde{F} + \mathsf{div}\left(\tilde{F} \otimes \mathbf{u}\right) + \mathsf{grad}\left(\frac{1}{2} g \frac{\tilde{h}}{\phi(x)}\right) &= 0 \end{aligned}$$

where $\tilde{h} = \phi(x)h$, $\tilde{F} = \tilde{h}\mathbf{u}$

Volume penalization of shallow water equations

Euler–Poincaré theory

Applying Hamilton's principle of least action to
$\mathcal{L} = \int \frac{1}{2} h(|\mathbf{u}|^2 - gh)\phi \, \mathrm{d}x \, \mathrm{d}y \, \mathrm{d}t$ gives

$$\begin{aligned} \frac{\partial}{\partial t}\tilde{h} + \mathsf{div}\tilde{F} &= 0 \\ \frac{\partial}{\partial t}\tilde{F} + \mathsf{div}\left(\tilde{F} \otimes \mathbf{u}\right) + \mathsf{grad}\left(\frac{1}{2} g \frac{\tilde{h}}{\phi(x)}\right) &= -\frac{1}{\eta}\chi(x)\mathbf{u} \end{aligned}$$

where $\tilde{h} = \phi(x) h$, $\tilde{F} = \tilde{h}\mathbf{u}$; $\eta \ll 1$ is the permeability due to viscosity.

Volume penalization of shallow water equations

Accuracy and scaling of penalization

- Error in h is $O(\alpha)$ *(from reflectance at boundary)*.
- Error in $\mathbf{u}$ is $O(\eta^{1/2})$ *(from Navier–Stokes penalization)*.
- Method is $O(\Delta x)$ since $\Delta x \propto \eta^{1/2}$.

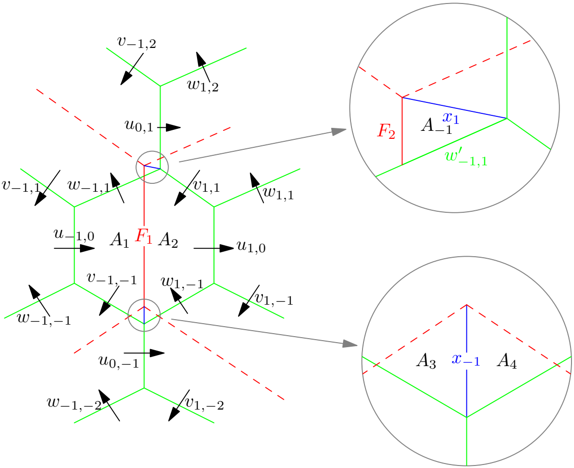
Volume penalization of shallow water equations

Accuracy and scaling of penalization

- Error in h is $O(\alpha)$ *(from reflectance at boundary)*.
- Error in $\mathbf{u}$ is $O(\eta^{1/2})$ *(from Navier–Stokes penalization)*.
- Method is $O(\Delta x)$ since $\Delta x \propto \eta^{1/2}$.

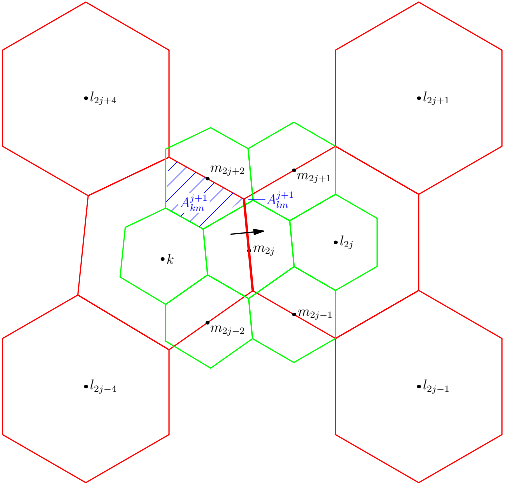
Use dynamic local grid refinement *(h-refinement)*.

Extension to icosahedral C-grid on sphere: flux restriction



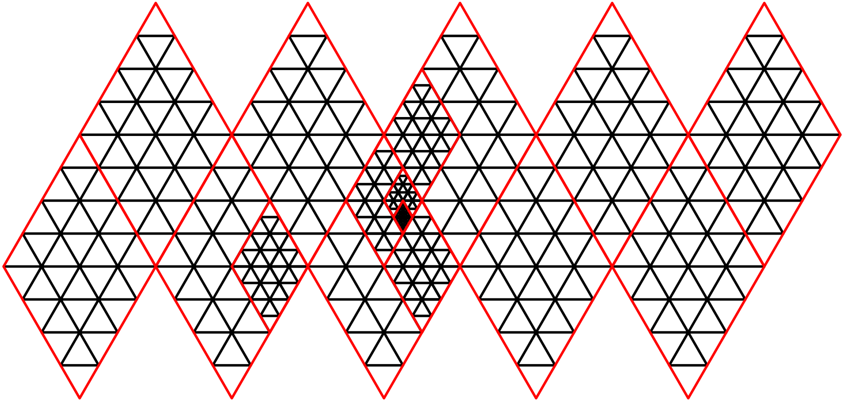
Small overlapping areas due to the non-uniform C-grid structure on the sphere.

Extension to icosahedral C-grid on sphere: flux restriction

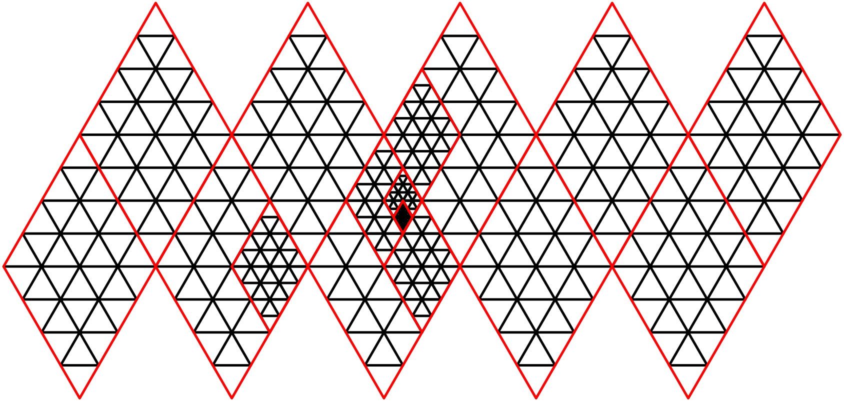


Fine and coarse scale cells to calculate flux restriction through coarse edge indicated by arrow. A_{km}^{j+1} and A_{lm}^{j+1} are partial areas.

Hybrid data structure: irregular tree data structure with regular patches

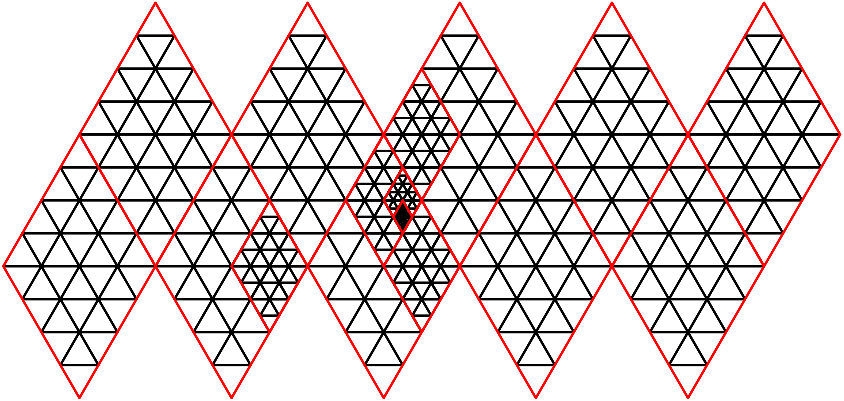


Hybrid data structure: irregular tree data structure with regular patches



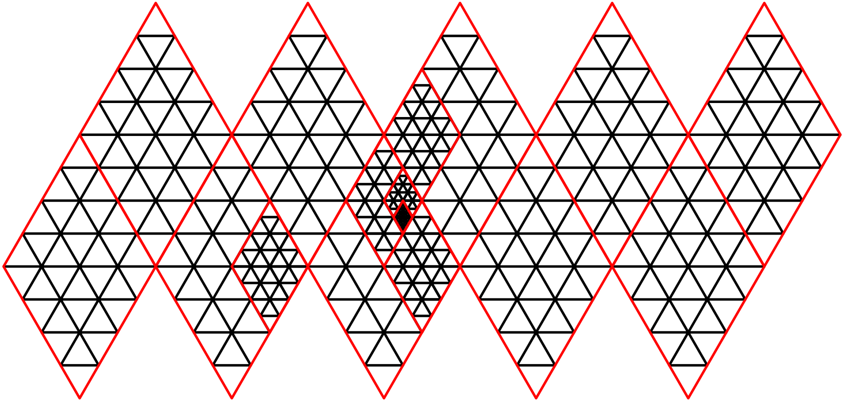
- Icosahedron divided into 10 regular lozenge domains.

Hybrid data structure: irregular tree data structure with regular patches



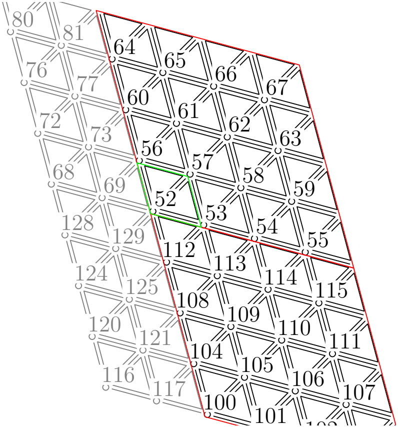
- Icosahedron divided into 10 regular lozenge domains.
- Domains refined adaptively into sub-domains.

Hybrid data structure: irregular tree data structure with regular patches



- Icosahedron divided into 10 regular lozenge domains.
- Domains refined adaptively into sub-domains.
- Lowest level locally is regular 4×4 patch.

Computational grid with ghost cells



4×4 patch is regular grid of elements. Element is one node, two triangles and three edges. Ghost points added at edges of sub-domain.

Parallelization

- Sub-domains distributed to different cores.
- Ghost points added and values communicated as necessary for operators.
- `Metis` graph partitioner improves load balancing.
- Communications occur at each trend computation and at each grid adaptation step.
- Where possible communication is non-blocking.

Results

Grid resolution

J	N	d.o.f	Δx $[km]$	T
5	10,242	40,962	239.8	51
6	40,962	163,842	119.9	101
7	163,842	655,362	60.0	202
8	655,362	2,621,442	30.0	404
9	2,621,442	10,485,762	15.0	809
10	10,485,762	41,943,042	7.5	1619
11	41,943,042	167,772,162	3.7	3238
12	167,772,162	671,088,642	1.9	6476

Grid resolution

J	N	d.o.f	Δx $[km]$	T
5	10,242	40,962	239.8	51
6	40,962	163,842	119.9	101
7	163,842	655,362	60.0	202
8	655,362	2,621,442	30.0	404
9	2,621,442	10,485,762	15.0	809
10	10,485,762	41,943,042	7.5	1619
11	41,943,042	167,772,162	3.7	3238
12	167,772,162	671,088,642	1.9	6476

- Optimize coarse grid, e.g. $J = 5$ *(Xu 2006; Heikes & Randall 1995)*.

Grid resolution

J	N	d.o.f	Δx $[km]$	T
5	10,242	40,962	239.8	51
6	40,962	163,842	119.9	101
7	163,842	655,362	60.0	202
8	655,362	2,621,442	30.0	404
9	2,621,442	10,485,762	15.0	809
10	10,485,762	41,943,042	7.5	1619
11	41,943,042	167,772,162	3.7	3238
12	167,772,162	671,088,642	1.9	6476

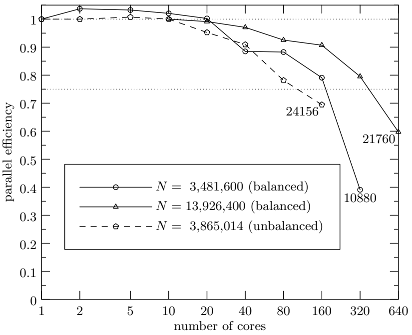
- Optimize coarse grid, e.g. $J = 5$ *(Xu 2006; Heikes & Randall 1995)*.
- Finer grids by recursive edge-bisection, e.g. $j = 6, 7, 8, 9, 10, \dots$

Grid resolution

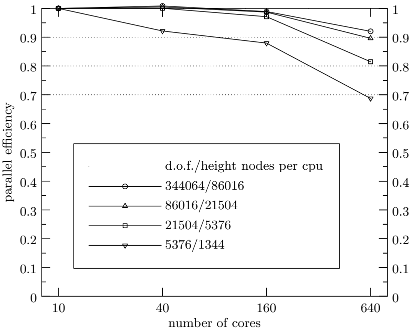
J	N	d.o.f	Δx $[km]$	T
5	10,242	40,962	239.8	51
6	40,962	163,842	119.9	101
7	163,842	655,362	60.0	202
8	655,362	2,621,442	30.0	404
9	2,621,442	10,485,762	15.0	809
10	10,485,762	41,943,042	7.5	1619
11	41,943,042	167,772,162	3.7	3238
12	167,772,162	671,088,642	1.9	6476

- Optimize coarse grid, e.g. $J = 5$ *(Xu 2006; Heikes & Randall 1995)*.
- Finer grids by recursive edge-bisection, e.g. $j = 6, 7, 8, 9, 10, \ldots$
- Local adaptive grid scale controlled by error tolerance ε.

Parallel scaling



Strong efficiency



Weak efficiency

Computational performance and adaptive overhead

Computational performance and adaptive overhead

- 5.0 times slower per active node than non-adaptive pseudo-spectral solver `swbob`.

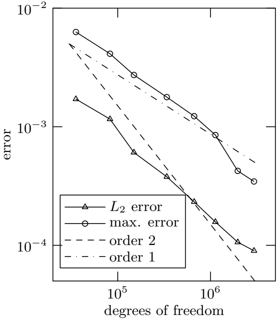
Computational performance and adaptive overhead

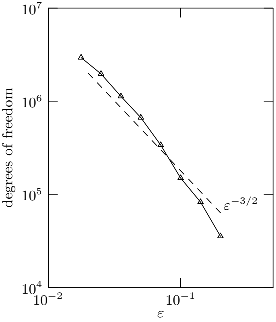
- 5.0 times slower per active node than non-adaptive pseudo-spectral solver `swbob`.
- 3.4 times slower per active node than non-adaptive TRiSK.

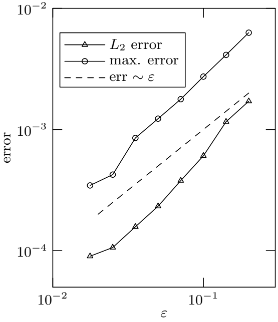
Computational performance and adaptive overhead

- 5.0 times slower per active node than non-adaptive pseudo-spectral solver `swbob`.
- 3.4 times slower per active node than non-adaptive TRiSK.
- Overall code is 3 to 10 times faster than pseudo-spectral and 4 to 15 times faster than non-adaptive TRiSK due to compression.

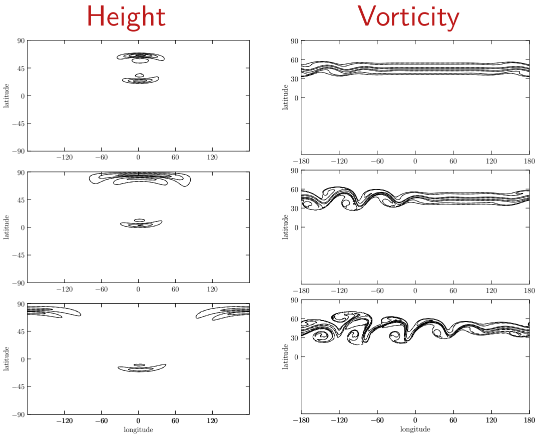
Williamson test case 2: error control







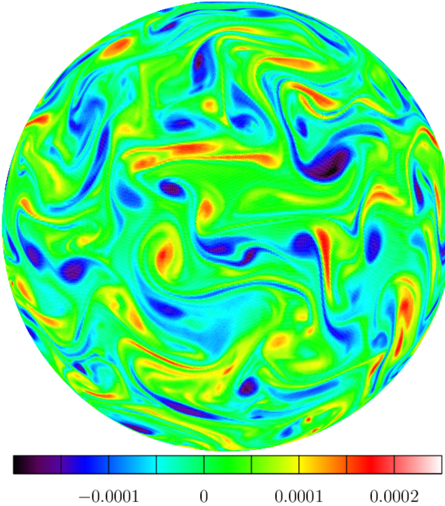
Unstable zonal jet on the sphere *(Galewsky et al. 2004)*



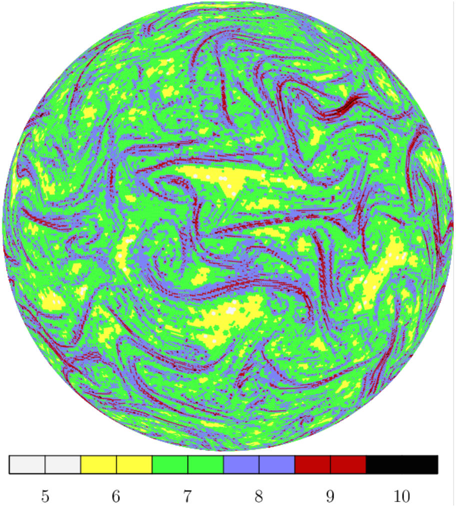
Tolerance $\epsilon = 5 \times 10^{-3}$ and $J = 9$. Height perturbation at 2, 4 and 6 hours and relative vorticity at 4, 5 and 6 days. (- - -) is non-adaptive $J = 10$ reference simulation, but results are mostly indistinguishable.

Unstable zonal jet on the sphere *(Galewsky et al. 2004)*

Viscous shallow water turbulence



Relative vorticity

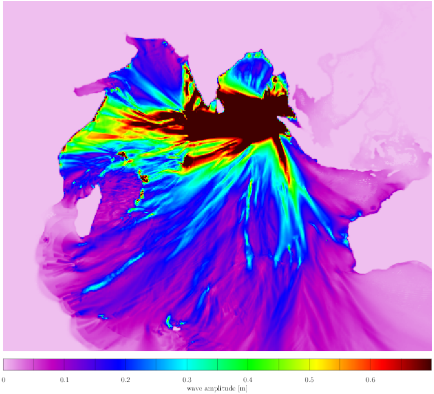


Level

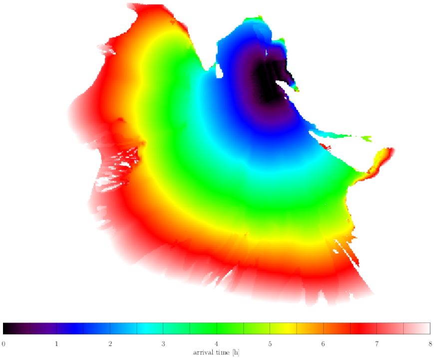
2004 Indonesian tsunami: 1.9 km resolution

2004 Indonesian tsunami: 1.9 km resolution

Max wave height



Arrival time (≥ 6 *cm wave*)



Conclusions

Spherical wavelets for adaptivity

- Multiscale representation
- Dynamic adaptivity controlled by local error estimate *or* static nesting
- Adaptivity overlay on existing TRiSK discretization
- Hybrid data structure
- Efficient parallelization using `mpi` and `metis`.
- Volume penalization for coastlines in ocean model

Conclusions

Spherical wavelets for adaptivity

- Multiscale representation
- Dynamic adaptivity controlled by local error estimate *or* static nesting
- Adaptivity overlay on existing TRiSK discretization
- Hybrid data structure
- Efficient parallelization using `mpi` and `metis`.
- Volume penalization for coastlines in ocean model

Future work

- 3-D hydrostatic extension, subgrid parameterizations

Conclusions

Spherical wavelets for adaptivity

- Multiscale representation
- Dynamic adaptivity controlled by local error estimate *or* static nesting
- Adaptivity overlay on existing TRiSK discretization
- Hybrid data structure
- Efficient parallelization using `mpi` and `metis`.
- Volume penalization for coastlines in ocean model

Future work

- 3-D hydrostatic extension, subgrid parameterizations

Details

Q.J.R. Meteorol. Soc. (2013) `DOI:10.1002/qj.2097`

arxiv.org/abs/1404.0405