Numerical Methods for Climate Models

Dr Hilary Weller <h.weller@reading.ac.uk>
Lecturer Version

13-16 September, 2021

NCAS and University of Reading
Climate Modelling Summer School

Reading and Resources

Python:

- "A Hands-On Introduction to Using Python in the Atmospheric and Oceanic Sciences" by Johnny Wei-Bing Lin. Download for free or buy cheaply from `http://www.johnny-lin.com/pyintro`.
- "Think Python. How to Think Like a Computer Scientist" by Allen B. Downey. Download from free from `https://greenteapress.com/wp/think-python-2e/`
- "Numerical Methods in Engineering with Python 3" by Jaan Kiusalaas `https://doi.org/10.1017/CBO9781139523899`
- `http://www.python.org`
- `http://docs.python.org/tutorial/`
- `https://matplotlib.org/stable/users/index.html`
- `http://learnpythonthehardway.org/`
- `http://www.ibiblio.org/g2swap/byteofpython/read/`
- "Making Use of Python" by Rashi Gupta
- "Python Essential Reference" by David M. Beazley (Addison Wesley)
- "Learning Python" Mark Lutz (O'Reilly Media)
- Free courses in Python and other languages `https://www.codecademy.com/`

Numerical Methods:

- Wikipedia `http://en.wikipedia.org`
- Durran, D. R. Numerical methods for fluid dynamics (Springer).
- LeVeque, R. Numerical Methods for Conservation Laws (Springer)
- Ortega, J.M. and Poole, W.G. An Introduction to Numerical Methods for Differential Equations. 1981 (Pitman Publishing Inc)
- Ferziger, J. H. and Peric, M. Computational Methods for Fluid Dynamics (Springer).
- Numerical Recipes: The Art of Scientific Computing, Third Edition (2007), (Cambridge University Press). `http://numerical.recipes/`
- Lloyd N. Trefethen. Finite Difference and Spectral Methods for Ordinary and Partial Differential Equations. `https://people.maths.ox.ac.uk/trefethen/pdetext.html`

Contents

Chapter 1: Linear Advection

1.1 The Navier Stokes Equations

The Navier-Stokes Equations for a compressible, rotating atmosphere

Momentum	$\dfrac{D\mathbf{u}}{Dt} = \text{-}2\boldsymbol{\Omega}\times\mathbf{u} - \dfrac{\nabla p}{\rho} + \mathbf{g} + \mu_u\left(\nabla^2\mathbf{u} + \dfrac{1}{3}\nabla(\nabla\cdot\mathbf{u})\right)$
Continuity	$\dfrac{D\rho}{Dt} + \rho\nabla\cdot\mathbf{u} = 0$
Potential temperature	$\dfrac{D\theta}{Dt} = Q + \mu_\theta\nabla^2\theta$

An equation of state, eg perfect gas law, $p = \rho RT$

Where the Lagrangian derivative is defined as $\dfrac{D\phi}{Dt} = \dfrac{\partial\phi}{\partial t} + \mathbf{u}\cdot\nabla\phi$

$\mathbf{u}$	Wind vector	$\mathbf{g}$	Gravity vector (downwards)
t	Time	θ	Potential temperature, $T\,(p_0/p)^\kappa$
$\boldsymbol{\Omega}$	Rotation rate of planet	κ	heat capacity ratio ≈ 1.4
ρ	Density of air	Q	Source of heat
p	Atmospheric pressure	μ_u, μ_θ	Diffusion coefficients
ϕ	Any atmospheric constituent		

Each of these equations has an advection term in the Lagrangian derivative.

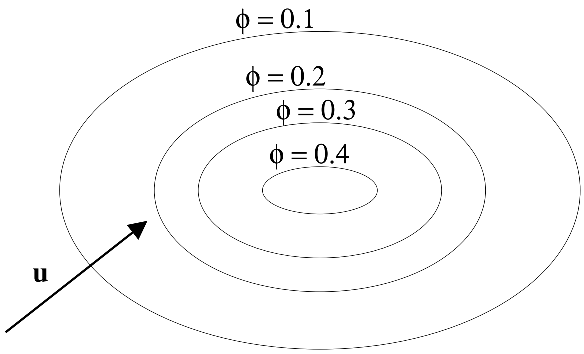
What does $\mathbf{u}\cdot\nabla\phi$ mean?

1.2 The Lagrangian Derivative and Advection

All of the NS equations include advection – properties of the atmosphere are moved by the wind. Consider a property ϕ (which could be wind, density or temperature). Advection with no other sources of change is governed by:

$$\frac{D\phi}{Dt} = \frac{\partial\phi}{\partial t} + \mathbf{u}\cdot\nabla\phi = 0 \tag{1.1}$$

Changes of ϕ are produced by the component of the wind in the same direction as gradients of ϕ. In order to understand why the $\mathbf{u}\cdot\nabla\phi$ term leads to changes in ϕ, consider a region of polluted atmosphere where the pollutant has the concentration contours shown below:



Exercise: Draw on the figure the directions of the gradients of ϕ and thus mark with a $+$, $-$ or 0 locations where $\mathbf{u}\cdot\nabla\phi$ is positive, negative and zero. Thus deduce where ϕ is increasing, decreasing or staying the same based on equation 1.1. Hence overlay contours of ϕ at a later time.

$\mathbf{u}\cdot\nabla\phi$ is the linear advection advection term – ϕ being moved around by the wind.

$\mathbf{u}\cdot\nabla\mathbf{u}$ is non-linear advection – the wind being moved around by the wind.

1.3 Advection and Diffusion

$$\underset{\substack{\text{Lagrangian}\\ \text{derivative}}}{\frac{D\phi}{Dt}} = \underset{\substack{\text{Rate of change}\\ \text{at fixed point}}}{\frac{\partial\phi}{\partial t}} + \underset{\substack{\text{Advection}\\ \text{of } \phi}}{\mathbf{u}\cdot\nabla\phi} = \underset{\substack{\text{Sources}\\ \text{and sinks}}}{S} + \underset{\substack{\text{Diffusion}\\ \text{of } \phi}}{\mu_\Psi \nabla^2\phi}$$

1.4 Properties of Advection

Which of the following are true about advection?

1. Advection moves fields around without changing their shape. False. Fields can be deformed.
2. Advection mixes areas of high and low concentration. Not quite. Diffusion is needed for mixing. Although advection can lead to intermingling
3. Nothing can be created or destroyed by advection. True. Advection is conservative
4. Advection moves fields from areas of high concentration to low concentration. False. This is diffusion. Advection moves fields in the direction of the wind
5. Advection cannot create negative values of a field that is initially all positive. True. New extrema cannot be generated. If you are at an extrema then the gradient is zero so the rate of change is also zero. This also means that initial maxima and minima should keep the same value.
6. Non-linear advection is not bounded. False. New extrema cannot be generated. If you are at an extrema then the gradient is zero so the rate of change is also zero. This also means that initial maxima and minima should keep the same value.

Exercise

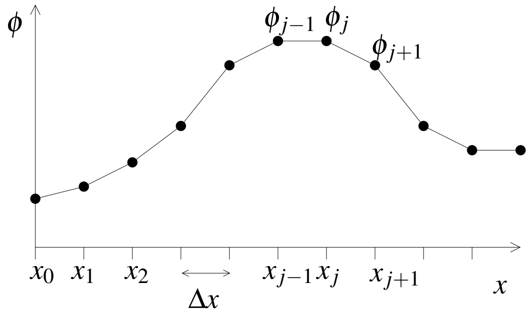
Assuming that ϕ varies only in the x direction and that the wind vector, $\mathbf{u} = \begin{pmatrix} u \\ v \\ w \end{pmatrix}$, write the one-dimensional linear advection equation for ϕ.

$$\frac{\partial\phi}{\partial t} + u\frac{\partial\phi}{\partial x} = 0$$

Chapter 2: Finite Differences

Partial differential equations (PDEs) such as the Navier-Stokes equations and linear advection equation can be solved approximately by splitting up space and time into discrete points and approximating the gradients using differences.

- For example, if we consider only one-dimensional space (only variations in the x direction), then we can divide the space between $x=0$ and $x=1$ into N equal intervals, each of size Δx, so that $x_j = j\Delta x$ for $j=1,2,\ldots,N$.
- We can divide time into time steps Δt, so that $t_n = n\Delta t,\ n=0,1,2,\ldots$.



- We wish to solve the 1D linear advection equation, $\frac{\partial\phi}{\partial t} + u\frac{\partial\phi}{\partial x} = 0$, where u is the known wind speed.
- Define $\phi_j^{(n)} = \phi(x_j,t_n)$, $\left.\frac{\partial\phi}{\partial x}\right._j^{(n)} = \frac{\partial\phi}{\partial x}(x_j,t_n)$, $\left.\frac{\partial\phi}{\partial t}\right._j^{(n)} = \frac{\partial\phi}{\partial t}(x_j,t_n)$.
- At time n and position j we can make the following approximations:

$$\left.\frac{\partial\phi}{\partial x}\right._j^{(n)} \approx \frac{\phi_{j+1}^{(n)} - \phi_{j-1}^{(n)}}{2\Delta x} \qquad \left.\frac{\partial\phi}{\partial t}\right._j^{(n)} \approx \frac{\phi_j^{(n+1)} - \phi_j^{(n-1)}}{2\Delta t}$$

- These can be substituted into the linear advection equation to give

$$\frac{\phi_j^{(n+1)} - \phi_j^{(n-1)}}{2\Delta t} + u\frac{\phi_{j+1}^{(n)} - \phi_{j-1}^{(n)}}{2\Delta x} = 0$$

- This can be re-arranged to give $\phi_j^{(n+1)}$, ϕ_j at the next time-step as a function of ϕs at previous time-steps and at adjacent locations. This can be simplified using the Courant number, $c = u\Delta t/\Delta x$ to give:

$$\phi_j^{(n+1)} = \phi_j^{(n-1)} - c_j\left(\phi_{j+1}^{(n)} - \phi_{j-1}^{(n)}\right)$$

- This is the CTCS scheme - Centred in Time, Centred in space, because the approximations for $\partial\phi/\partial t$ and $\partial\phi/\partial x$ are both centred.
- CTCS uses three time levels (two previous time levels). However at the first time step, only one previous time level is available. Therefore a forward in time approximation is needed for $\left.\frac{\partial\phi}{\partial t}\right._j^{(n)}$:

$$\left.\frac{\partial\phi}{\partial t}\right._j^{(n)} \approx \frac{\phi_j^{(n+1)} - \phi_j^{(n)}}{\Delta t}$$

Combined with centred in space, this gives the FTCS scheme:

$$\phi_j^{(n+1)} = \phi_j^{(n)} - \frac{c_j}{2}\left(\phi_{j+1}^{(n)} - \phi_{j-1}^{(n)}\right)$$

- Python code implementing CTCS is given in section 2.1

2.1 Python Code to Solve the Linear Advection Equation using CTCS

```
import numpy as np       # External library for numerical calculations
import matplotlib.pyplot as plt   # Plotting library

def initialBell(x):
    "Initial conditions as a function of space, x"
    return np.where(x%1. < 0.5, np.power(np.sin(2*x*np.pi), 2), 0)

# Setup space, x, initial phi profile and Courant number
nx = 40                   # number of points in space
c = 0.2                   # The Courant number
x = np.linspace(0.0, 1.0, nx+1) # From zero to one inclusive
nt = 40                   # The number of time steps
phi = initialBell(x)      # Three time levels of the dependent variable,
phiNew = phi.copy()       # phi
phiOld = phi.copy()

# derived or assumed quantities
u = 1.0
dx = 1./nx
dt = c*dx/u

# Plot the initial conditions
plt.plot(x, phi, 'k', label='initial conditions')
plt.legend(loc='best')
plt.ylabel('$\phi$')
plt.axhline(0, linestyle=':', color='black')
plt.pause(1)
```



```
# FTCS for the first time-step, looping over space
for j in range(1,nx):        # loops from 1 to nx-1
    phi[j] = phiOld[j] - 0.5*c*(phiOld[j+1] - phiOld[j-1])
# apply periodic boundary conditions
phi[0] = phiOld[0] - 0.5*c*(phiOld[1] - phiOld[nx-1])
phi[nx] = phi[0]

# Loop over remaining time-steps (nt-1) using CTCS
for n in range(1,nt):        # loop from 1 to nt-1
    for j in range(1,nx):    # loop over space from 1 to nx-1
        phiNew[j] = phiOld[j] - c*(phi[j+1] - phi[j-1])
    # apply periodic boundary conditions
    phiNew[0] = phiOld[0] - c*(phi[1] - phi[nx-1])
    phiNew[nx] = phiNew[0]
    # update phi for the next time-step
    phiOld = phi.copy()
    phi = phiNew.copy()
    # Replot
    plt.cla()
    plt.plot(x, initialBell(x - u*n*dt), 'k', label='analytic')
    plt.plot(x, phi, 'b', label='CTCS')
    plt.legend(loc='best')
    plt.ylabel('$\phi$')
    plt.axhline(0, linestyle=':', color='black')
    plt.pause(0.05)

plt.show() # To keep the plot showing at the end
```

2.2 Exercises

1. Find a finite difference formula for the second derivative, $\partial^2\phi/\partial x^2$, on a grid with spacing Δx indexed by j.
2. Hence derive the forward in time, centred in space (FTCS) scheme for the diffusion equation:
$$\frac{\partial \phi}{\partial t} = K\frac{\partial^2 \phi}{\partial x^2}.$$
3. The equation for inertial oscillations given in section 3.2.1 is:
$$\frac{\partial \mathbf{u}}{\partial t} = \text{-}2\mathbf{\Omega}\times\mathbf{u}.$$
   (a) Write equations for the horizontal components of **u**, assuming that:
$$\mathbf{u} = \begin{pmatrix} u \\ v \\ w \end{pmatrix}, \quad 2\mathbf{\Omega} = \begin{pmatrix} 0 \\ 0 \\ f \end{pmatrix}.$$
   (b) Hence write down a numerical method for integrating u and v forward in time.
   (c) From u and v, write down equations for calculating the location of a parcel of air, $(x^{(n)}, y^{(n)})$ at time step n from the previous time step.
4. Derive a finite difference scheme for Burger's equation in one dimension:
$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = 0.$$

2.2.1 Answers

1. $\dfrac{\partial^2\phi}{\partial x^2}_j \approx \dfrac{\phi_{j+1} - 2\phi_j + \phi_{j-1}}{\Delta x^2}$
2. $\dfrac{\phi_j^{(n+1)} - \phi_j^{(n)}}{\Delta t} \approx \dfrac{\phi_{j+1}^{(n)} - 2\phi_j^{(n)} + \phi_{j-1}^{(n)}}{\Delta x^2}$
3. $\partial u/\partial t = fv$, $\partial v/\partial t = -fu$
   $u^{(n+1)} = u^{(n)} + \Delta t\ f\ v^{(n)}$
   $v^{(n+1)} = v^{(n)} - \Delta t\ f\ u^{(n+1)}$
   $x^{(n+1)} = x^{(n)} + \Delta t\ u^{(n)}$
   $y^{(n+1)} = y^{(n)} + \Delta t\ v^{(n)}$
4. $u_j^{(n+1)} = u_j^{(n)} - \Delta t\ u_j^{(n)}\dfrac{u_j^{(n)} - u_{j-1}^{(n)}}{\Delta x}$

2.3 Python Code to Solve the Inertial Oscillation Equation using forward-backward time-stepping

```
# Numerical solution of the inertial osicllation equation using
# forward-backward time-stepping with Coriolis parameter f
# dudt = f*v
# dvdt = -f*u

import numpy as np        # External library for numerical calculations
import matplotlib.pyplot as plt   # Plotting library

# Setup parameters
f = 1e-4                  # Coriolis parameter
nt = 12                   # Number of time steps
dt = 5000                 # Time step in seconds
# Initial conditions (in meters or m/s)
x0 = 0.
y0 = 1e5
u0 = 10.
v0 = 0.
# Initialise velocity from initial conditions
u = u0
v = v0
# Store all locations for plotting and store initial locations
x = np.zeros(nt+1)
y = np.zeros(nt+1)
x[0] = x0
y[0] = y0
```



```
# Loop over all time-steps
for n in range(nt):
    u += dt*f*v
    v -= dt*f*u
    x[n+1] = x[n] + dt*u
    y[n+1] = y[n] + dt*v

# Analytic solution for the location as a function of time
times = np.linspace(0,nt*dt, nt+1)
xa = x0 + 1/f*(u0*np.sin(f*times) - v0*np.cos(f*times) + v0)
ya = y0 + 1/f*(u0*np.cos(f*times) + v0*np.sin(f*times) - u0)

# Plot the solution in comparison to the analytic solution
plt.plot(xa, ya, '-k+', label='analytic')
plt.plot(x, y, '-bo', label='forward-backward')
plt.legend(loc='best')
plt.xlabel('x')
plt.ylabel('y')
plt.show()
```

2.4 Practical: Solving a Differential Equation using Finite Differences

This practical is open ended and we will build on it during the week. Everyone has different knowledge and experience coding numerical methods so during this practical you will choose your own equation to solve and choose or design a numerical method to solve it. If you have not coded numerical methods before, you might like to start with the one dimensional linear advection equation and make modifications to the code in section 2.1 or inertial oscillations and start from the code in section 2.3. For the more adventurous, you could choose an equation that you are interested in or pick a couple of terms from one of the Navier-Stokes equations to model. Please discuss your decisions with the staff. Also feel free to find equations and numerical methods online. Some equations that you might like to consider:

- John Methven's Potential Vorticity Problem
- The linear advection equation (this is not easy to solve well)
- The non-linear advection equation (Burger's equation)
- The linear or non-linear shallow-water equations
- The diffusion equation (easier than the linear advection equation)
- Inertial oscillations. You could start from the code in section 2.3, see what happens when you make the time-step very large and then try to make the time-stepping implicit.

You will probably learn most if you stick with one dimensional equations.

You will have an opportunity at the end of the week to tell the rest of the class what equation you have chosen, how you have chosen to solve it and if you have any results or problems.

Chapter 3: The Navier Stokes Equations

The Navier-Stokes Equations for a compressible, rotating atmosphere

Momentum $$\frac{D\mathbf{u}}{Dt} = \text{-}2\mathbf{\Omega}\times\mathbf{u} - \frac{\nabla p}{\rho} + \mathbf{g} + \mu_u\left(\nabla^2\mathbf{u} + \frac{1}{3}\nabla(\nabla\cdot\mathbf{u})\right)$$

Continuity $$\frac{D\rho}{Dt} + \rho\nabla\cdot\mathbf{u} = 0$$

Potential temperature $$\frac{D\theta}{Dt} = Q + \mu_\theta\nabla^2\theta$$

An equation of state, eg perfect gas law, $p = \rho RT$

Where the Lagrangian derivative is defined as $\dfrac{D\Psi}{Dt} = \dfrac{\partial\Psi}{\partial t} + \mathbf{u}\cdot\nabla\Psi$

$\mathbf{u}$	Wind vector	$\mathbf{g}$	Gravity vector (downwards)
t	Time	θ	Potential temperature, $T\,(p_0/p)^\kappa$
$\mathbf{\Omega}$	Rotation rate of planet	κ	heat capacity ratio ≈ 1.4
ρ	Density of air	Q	Source of heat
p	Atmospheric pressure	μ_u, μ_θ	Diffusion coefficients

- What does $\nabla\cdot\mathbf{u}$ mean?
- What does ∇p mean?
- What does $\mathbf{u}\cdot\nabla\Psi$ mean?
- What does $\mathbf{\Omega}\times\mathbf{u}$ mean?

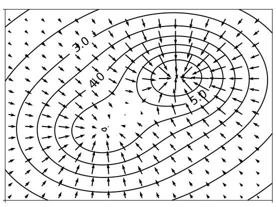
3.1 Vector Calculus Revision

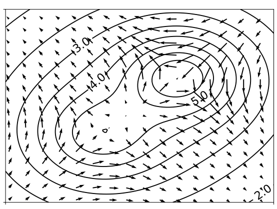
3.1.1 Gradients and Scalar Products

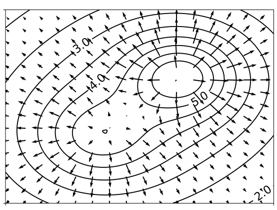
For a scalar quantity ϕ (which could be density, temperature or concentration of a pollutant), the gradient of ϕ is a vector:

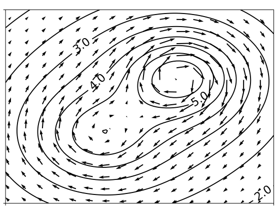
$$\nabla\phi = \begin{pmatrix} \dfrac{\partial\phi}{\partial x} \\ \dfrac{\partial\phi}{\partial y} \\ \dfrac{\partial\phi}{\partial z} \end{pmatrix}$$

Question: If ϕ is shown by the contours, which vector field shows $\nabla\phi$?

a  ✓

b  ×

c  ×

d  ×

Quiz [responses]

3.1.2 Dot Products

If the wind speed is a vector, $\mathbf{u} = \begin{pmatrix} u \\ v \\ w \end{pmatrix}$, then the dot product between vectors $\mathbf{u}$ and $\nabla\phi$, $\mathbf{u}\cdot\nabla\phi$, is (select all that apply):

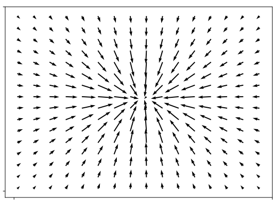
1. $\begin{pmatrix} u\frac{\partial\phi}{\partial x} \\ v\frac{\partial\phi}{\partial y} \\ w\frac{\partial\phi}{\partial z} \end{pmatrix}$ ×
2. $u\frac{\partial\phi}{\partial x} + v\frac{\partial\phi}{\partial y} + w\frac{\partial\phi}{\partial z}$ ✓
3. $\begin{pmatrix} v\frac{\partial\phi}{\partial z} - w\frac{\partial\phi}{\partial y} \\ -u\frac{\partial\phi}{\partial z} + w\frac{\partial\phi}{\partial x} \\ u\frac{\partial\phi}{\partial y} - v\frac{\partial\phi}{\partial x} \end{pmatrix}$ ×
4. $u\frac{\partial\phi}{\partial y} - v\frac{\partial\phi}{\partial z} + w\frac{\partial\phi}{\partial x}$ ×
5. $|\mathbf{u}||\nabla\phi|\cos\theta$ where θ is the angle between $\mathbf{u}$ and $\nabla\phi$ ✓
6. Zero when $\mathbf{u}$ and $\nabla\phi$ are parallel ×
7. Zero when $\mathbf{u}$ and $\nabla\phi$ are at right angles ✓
8. Negative when $\mathbf{u}$ and $\nabla\phi$ are in opposite directions.✓

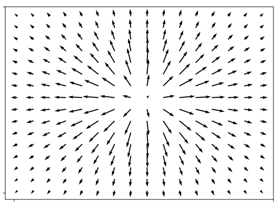
Quiz [responses]

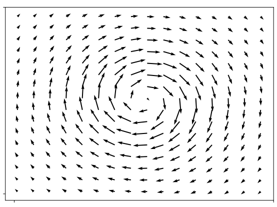
3.1.3 Divergence

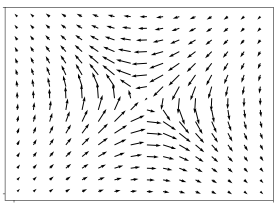
The divergence of a vector, $\mathbf{u} = \begin{pmatrix} u \\ v \\ w \end{pmatrix}$ is $\nabla\cdot\mathbf{u} = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}$.

Do these vector fields have positive, negative or near zero divergence? Quiz responses

a  +

b  -

c  0

d  0

Is the divergence in the atmosphere large or small? Small in 2D. Very small in 3D.

3.1.4 Laplacian

The Laplacian of a scalar field, ϕ is $\nabla^2\phi = \nabla\cdot(\nabla\phi)$. In component form, the Laplacian of ϕ is:

1. $\dfrac{\partial^2\phi}{\partial x^2}+\dfrac{\partial^2\phi}{\partial y^2}+\dfrac{\partial^2\phi}{\partial z^2}$ ✓
2. $\begin{pmatrix} \frac{\partial^2\phi}{\partial x^2} & \frac{\partial^2\phi}{\partial x\partial y} & \frac{\partial^2\phi}{\partial x\partial z} \\ \frac{\partial^2\phi}{\partial x\partial y} & \frac{\partial^2\phi}{\partial y^2} & \frac{\partial^2\phi}{\partial y\partial z} \\ \frac{\partial^2\phi}{\partial x\partial y} & \frac{\partial^2\phi}{\partial y\partial z} & \frac{\partial^2\phi}{\partial z^2} \end{pmatrix}$ ×
3. $\dfrac{\partial\phi}{\partial x}+\dfrac{\partial\phi}{\partial y}+\dfrac{\partial\phi}{\partial z}$ ×
4. $\begin{pmatrix} \frac{\partial\phi}{\partial x} \\ \frac{\partial\phi}{\partial y} \\ \frac{\partial\phi}{\partial z} \end{pmatrix}$ ×

Quiz [responses]

3.1.5 Cross Product

If vectors $\mathbf{x} = \begin{pmatrix} a \\ b \\ c \end{pmatrix}$ and $\mathbf{y} = \begin{pmatrix} d \\ e \\ f \end{pmatrix}$ then the cross product is

$$\mathbf{x}\times\mathbf{y} = \begin{vmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ a & b & c \\ d & e & f \end{vmatrix} = \begin{pmatrix} bf-ce \\ -af+cd \\ ae-bd \end{pmatrix}$$

If the velocity vector is $\mathbf{u} = \begin{pmatrix} u \\ v \\ w \end{pmatrix}$ and the angular velocity of the domain is $\mathbf{\Omega} = \begin{pmatrix} 0 \\ 0 \\ 1/2f \end{pmatrix}$ then the Cartesian components of the Coriolis term, $-2\mathbf{\Omega}\times\mathbf{u}$, are:

1. $\begin{pmatrix} fv \\ -fu \\ 0 \end{pmatrix}$ ✓
2. $\begin{pmatrix} -fv \\ fu \\ 0 \end{pmatrix}$ ×
3. $\begin{pmatrix} fu \\ fv \\ 0 \end{pmatrix}$ ×
4. $\begin{pmatrix} -2fu \\ -2fv \\ 0 \end{pmatrix}$ ×

Quiz [responses]

3.2 The Momentum Equation

$$\underset{\text{Lagrangian derivative}}{\frac{D\mathbf{u}}{Dt}} = \underset{\text{Coriolis}}{-2\mathbf{\Omega}\times\mathbf{u}} - \underset{\text{Pressure gradient}}{\frac{\nabla p}{\rho}} + \underset{\text{Gravitational acceleration}}{\mathbf{g}} + \underset{\text{Diffusion}}{\mu_u\left(\nabla^2\mathbf{u}+\frac{1}{3}\nabla(\nabla\cdot\mathbf{u})\right)}$$

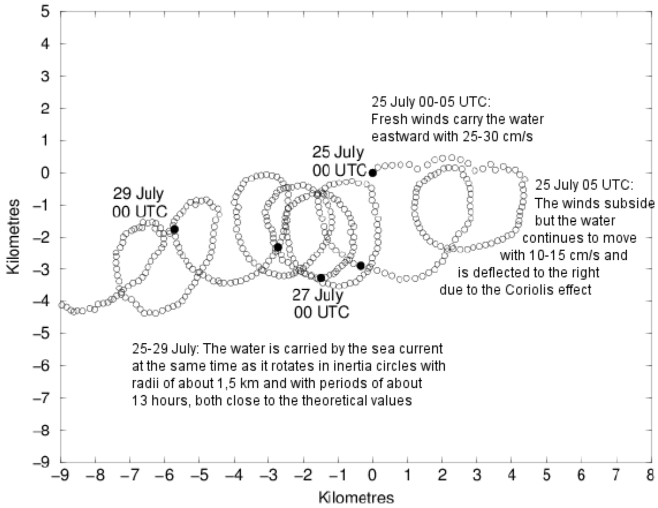
3.2.1 Coriolis

Inertial Oscillations governed by part of the momentum equation:

$$\frac{\partial \mathbf{u}}{\partial t} = -2\mathbf{\Omega}\times\mathbf{u}$$

- A drifting buoy set in motion by strong westerly winds in the Baltic Sea in July 1969.
- Once the wind subsides, the upper ocean follows inertia circles

Image copyright: Anders Persson.
Source of scientific data: Barry Broman.
Reproduced with kind permission from Anders Persson and Barry Broman.

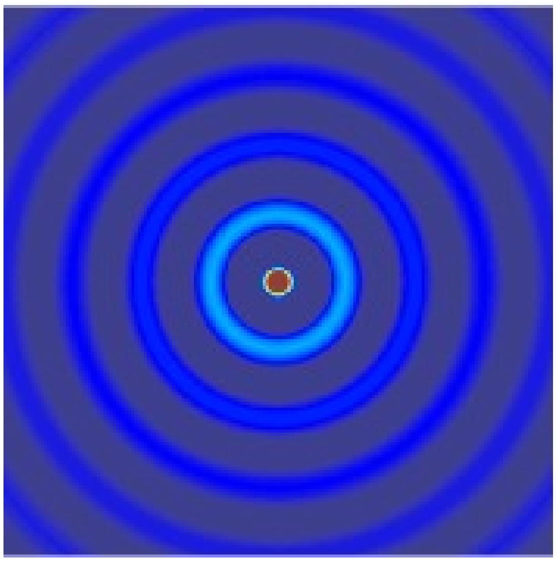


3.2.2 The Pressure Gradient Force

If the pressure gradient force is the only large term in the momentum equation, then together with the continuity equation and perfect gas law, we get equations for acoustic waves:

$$\begin{aligned}\frac{\partial \mathbf{u}}{\partial t} + \frac{1}{\rho_0}\nabla p &= 0\\ \frac{\partial p}{\partial t} + \rho_0 c^2 \nabla\cdot\mathbf{u} &= 0\end{aligned}$$

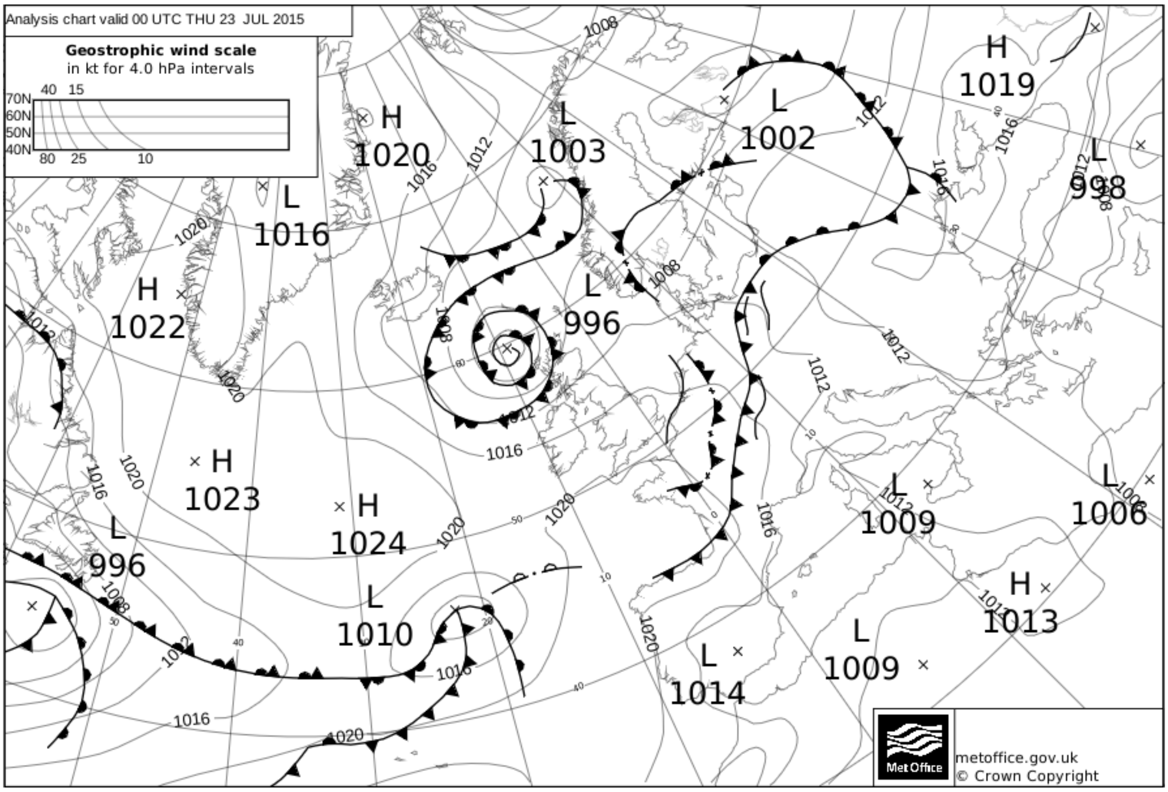
where ρ_0 is a reference density and c is the speed of sound.



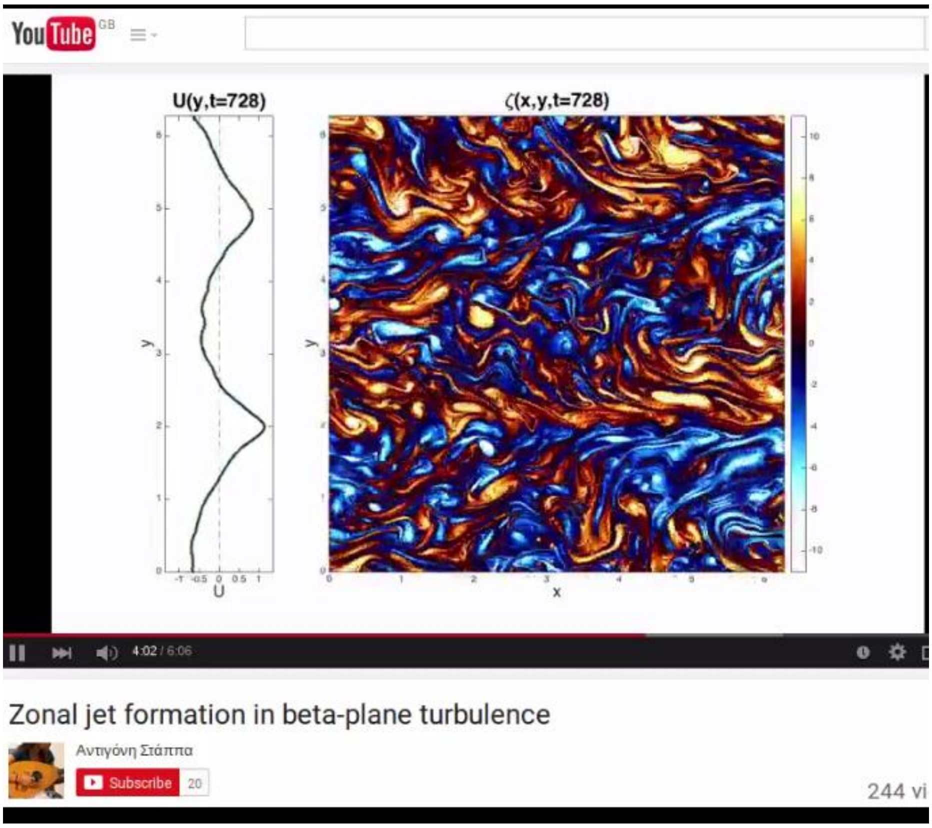
Pressure Gradients lead to very fast acceleration - Acoustic Waves

Geostrophic Balance: Pressure Gradients versus Coriolis

-2$\boldsymbol{\Omega} \times \mathbf{u} = \dfrac{\nabla p}{\rho}$. If $2\boldsymbol{\Omega} = \begin{pmatrix} 0 \\ 0 \\ f \end{pmatrix}$ and $\mathbf{u} = \begin{pmatrix} u \\ v \\ w \end{pmatrix}$ then $u = -\dfrac{1}{f\rho}\dfrac{\partial p}{\partial y}$, $v = \dfrac{1}{f\rho}\dfrac{\partial p}{\partial x}$.



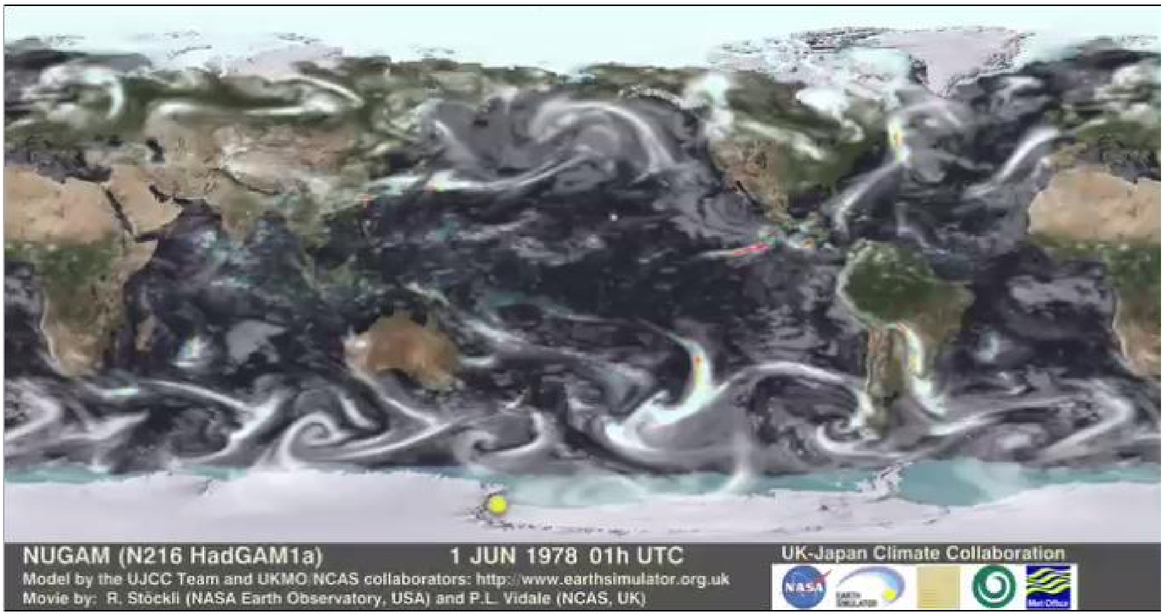
Geostrophic turbulence: pressure gradients, Coriolis and the (non-linear) advection of velocity by velocity



3.2.3 Gravitational Acceleration: Explosive Comulonimbus

3.2.4 The Complete Navier Stokes Equations

With moisture, phase changes, radiation, ... from NUGAM (courtesy of Pier Luigi)



Vector Calculus Revision: The Chain Rule

Two forms of the chain rule are

$$\frac{d(fg)}{dx} = f\frac{dg}{dx} + g\frac{df}{dx}$$

for scalar valued functions f and g and

$$\nabla\cdot(\phi\mathbf{u}) = \phi\nabla\cdot\mathbf{u} + \mathbf{u}\cdot\nabla\phi$$

for scalar valued ϕ and vector $\mathbf{u}$. There are other forms depending on the type of the variable and the type of the gradient, ie div, grad and curl.

3.3 Flux and Advective Form

The Navier-Stokes equations are usually derived by considering fluxes entering and leaving small volumes. This leads to the flux or conservative form of the advective term. For example, for the continuity equation this is

$$\frac{\partial\rho}{\partial t} + \nabla\cdot(\rho\mathbf{u}) = 0.$$

Exercise: Use the chain rule to show that this is equivalent to the form of the equation given at the beginning of this chapter.

$$\frac{\partial\rho}{\partial t} + \nabla\cdot(\rho\mathbf{u}) = \frac{\partial\rho}{\partial t} + \mathbf{u}\cdot\nabla\rho + \rho\nabla\cdot\mathbf{u} = \frac{D\rho}{Dt} + \rho\nabla\cdot\mathbf{u}$$

The equations at the beginning of the chapter can also be written in terms of density weighted variables such as $\rho\theta$.

Exercise (more difficult): Use the chain rule to find an equation for $\frac{\partial\rho\theta}{\partial t}$ in terms of $\frac{\partial\rho}{\partial t}$ and $\frac{\partial\theta}{\partial t}$. Then substitute in the expression for $\frac{\partial\rho}{\partial t}$ from the flux form of the continuity equation and the expression for $\frac{\partial\theta}{\partial t}$ to find an expression for $\frac{\partial\rho\theta}{\partial t}$. Hence use the chain rule to derive:

$$\frac{\partial\rho\theta}{\partial t}+\nabla\cdot(\rho\mathbf{u}\theta)=\rho Q+\rho\nabla\cdot\kappa_\theta\nabla\theta$$

$$\begin{aligned}\frac{\partial\rho\theta}{\partial t} &= \theta\frac{\partial\rho}{\partial t}+\rho\frac{\partial\theta}{\partial t}=\theta\{-\nabla\cdot(\rho\mathbf{u})\}+\rho\{-\mathbf{u}\cdot\nabla\theta+Q+\nabla\cdot(\kappa_\theta\nabla\theta)\}\\ &= -\{\theta\nabla\cdot(\rho\mathbf{u})+\rho\mathbf{u}\cdot\nabla\theta\}+\rho\{Q+\nabla\cdot(\kappa_\theta\nabla\theta)\}\\ &= -\nabla\cdot(\rho\mathbf{u}\theta)+\rho\{Q+\nabla\cdot(\kappa_\theta\nabla\theta)\}\end{aligned}$$

3.4 Simplifications of the Navier-Stokes Equations

These are the equations of motions as given at the beginning of this chapter:

$$\begin{aligned}\frac{D\mathbf{u}}{Dt} &= -2\mathbf{\Omega}\times\mathbf{u}-\frac{\nabla p}{\rho}+\mathbf{g}+\mu_u\left(\nabla^2\mathbf{u}+\frac{1}{3}\nabla(\nabla\cdot\mathbf{u})\right)\\ \frac{D\rho}{Dt}+\rho\nabla\cdot\mathbf{u} &= 0\\ \frac{D\theta}{Dt} &= Q+\mu_\theta\nabla^2\theta\end{aligned}$$

1. What does this simplification represent?

$$\begin{aligned}\frac{D\mathbf{u}}{Dt} &= -2\mathbf{\Omega}\times\mathbf{u}-\frac{\nabla p}{\rho}+\mathbf{g}+\mu_u\nabla^2\mathbf{u}\\ \nabla\cdot\mathbf{u} &= 0\end{aligned}$$

(a) Hydrostatic balance
(b) Inviscid (frictionless) flow
(c) Linear advection
(d) Constant density flow
(e) Divergence free flow
(f) Geostrophic balance
(g) Thermal wind
(h) Irrotational flow
(i) Adiabatic flow

Answers here Responses e (not necessarily d)

2. What does this simplification represent?

$$\begin{aligned} \frac{D\mathbf{u}}{Dt} &= \text{-}2\mathbf{\Omega} \times \mathbf{u} - \frac{\nabla p}{\rho} + \mathbf{g} \\ \frac{D\rho}{Dt} + \rho \nabla \cdot \mathbf{u} &= 0 \\ \frac{D\theta}{Dt} &= 0 \end{aligned}$$

(a) Hydrostatic balance
(b) Inviscid (frictionless) flow
(c) Linear advection
(d) Constant density flow
(e) Divergence free flow
(f) Geostrophic balance
(g) Thermal wind
(h) Irrotational flow
(i) Adiabatic flow

Answers here Responses b and i

3. What does this simplification represent?

$$2\mathbf{\Omega} \times \mathbf{u} = -\frac{\nabla p}{\rho}$$

(a) Hydrostatic balance
(b) Inviscid (frictionless) flow
(c) Linear advection
(d) Constant density flow
(e) Divergence free flow
(f) Geostrophic balance
(g) Thermal wind
(h) Irrotational flow
(i) Adiabatic flow

Answers here Responses f

4. What does this simplification represent?

$$-\frac{\nabla p}{\rho} + \mathbf{g} = 0$$

(a) Hydrostatic balance
(b) Inviscid (frictionless) flow
(c) Linear advection
(d) Constant density flow
(e) Divergence free flow
(f) Geostrophic balance
(g) Thermal wind
(h) Irrotational flow
(i) Adiabatic flow

Answers here Responses a

5. What does this simplification represent?

$$\frac{\partial \Psi}{\partial t} + \mathbf{u} \cdot \nabla \Psi = 0$$

(a) Hydrostatic balance
(b) Inviscid (frictionless) flow
(c) Linear advection
(d) Constant density flow
(e) Divergence free flow
(f) Geostrophic balance
(g) Thermal wind
(h) Irrotational flow
(i) Adiabatic flow

Answers here Responses c

More on the Practical

For this practical you have more choices:

1. Further analysis of the method that you have already chosen.
2. Implement a better numerical method for the equation you are solving.
3. Solve a different equation.

In particular, you may consider:

1. If you are solving an advection equation, you might like to consider using one of:
   (a) A monotonic advection scheme
   (b) Semi-Lagrangian advection
   (c) The Lax-Wendroff method
2. How do solution errors vary with resolution in time and space?
3. Is your numerical method stable for all time steps?
4. Is your numerical method conservative?
5. Is there any spurious behaviour in your solution? (eg unbounded results or unrealistic oscillations).
6. Compare explicit and implicit time-stepping.
7. You might like to consider using Runge-Kutta time-stepping
8. You might like to increase the order of accuracy of the spatial discretisation
9. If you are solving the shallow water equations
   (a) Try adding non-linear terms

   (b) Try using a different Arakawa grid
10. Try using variable spatial resolution or adaptive time-stepping

You should search online for ideas of a numerical method to use. You can also refer to my MSc teaching notes:
`http://www.met.reading.ac.uk/~sws02hs/teaching/MTMW12/MTMW12_2_lec.pdf`

Chapter 4: Spatial Discritisation for Atmospheric Modelling

Aim

- Describe a multitude of spatial discretisation methods used in atmospheric modelling
- Describe their advantages and disadvantages
- What modelling centres use what methods
- Different meshes of the sphere

Methods

- Finite difference
- Arakawa grids
- Semi-Lagrangian
- Finite Volume
- Finite Element
- Spectral Method
- Spectral Element
- Mixed finite element
- Discontinuous Galerkin

No spatial discretisation method is perfect. But some produce useful forecasts.

4.1 Shallow Water Equations

Non-linear

$$\frac{D\mathbf{u}}{Dt} = -2\Omega\times\mathbf{u} - g\nabla(h+h_0) \qquad (4.1)$$

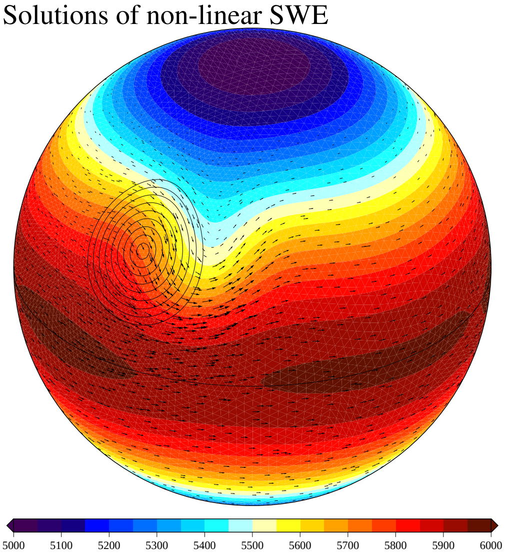
$$\frac{Dh}{Dt} + h\nabla\cdot\mathbf{u} = 0 \qquad (4.2)$$

Ω is rotation

Exercise: Linearise

Assume that $\mathbf{u}$ is small and that $h\to H+h$ where H is mean height and h is small (ignore mountain)

Solutions of non-linear SWE



Colour shows height of atmosphere.
Vectors show depth integrated wind.

$$\frac{\partial\mathbf{u}}{\partial t} = -2\Omega\times\mathbf{u} - g\nabla h \quad , \quad \frac{\partial h}{\partial t} + H\nabla\cdot\mathbf{u} = 0$$

Exercise: Write in one dimension, assuming variations only in the x direction

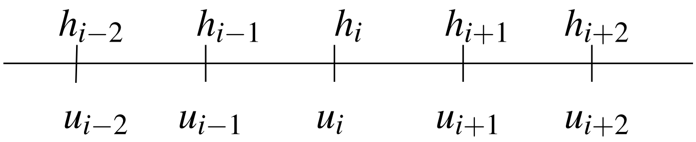
$$\frac{\partial u}{\partial t} = fv - g\frac{\partial h}{\partial x} \quad , \quad \frac{\partial h}{\partial t} + H\frac{\partial u}{\partial x} = 0$$

4.2 Finite Differences - Arakawa A-grid

To solve the linearised shallow water equations in one dimension:

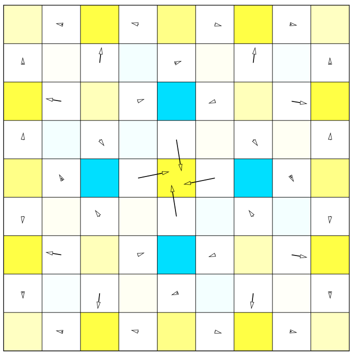
$$\frac{\partial u}{\partial t} = fv - g\frac{\partial h}{\partial x} \quad , \quad \frac{\partial h}{\partial t} + H\frac{\partial u}{\partial x} = 0$$

Store discrete values of h and u on a grid.



$$\left.\frac{\partial u}{\partial x}\right._j \approx \frac{u_{j+1} - u_{j-1}}{2\Delta x}$$
$$\left.\frac{\partial h}{\partial x}\right._j \approx \frac{h_{j+1} - h_{j-1}}{2\Delta x}$$

This can be extended to two dimensions and gives these solutions:



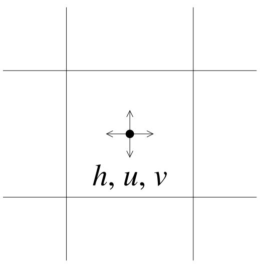
The Arakawa A-grid is not used by operational centres in this form.

4.3 Arakawa Grids [Arakawa and Lamb, 1977]

Consider the component form of the 2D linearised SWE:

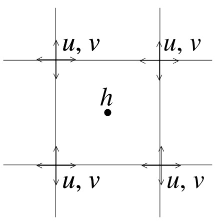
$$\frac{\partial u}{\partial t} = fv - g\frac{\partial h}{\partial x} \tag{4.3}$$
$$\frac{\partial v}{\partial t} = -fu - g\frac{\partial h}{\partial y} \tag{4.4}$$
$$\frac{\partial h}{\partial t} = -H\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right) \tag{4.5}$$

A-grid



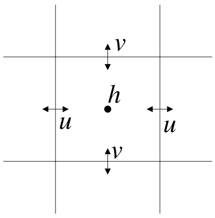
- Non-compact discretisation of $\partial u/\partial x$, $\partial u/\partial y$, $\partial h/\partial x$ and $\partial h/\partial y$ leads to spurious gravity modes.
- Grid-scale oscillations
- Simple
- Acurate representation of inertial oscillations (u and v are together)

B-grid



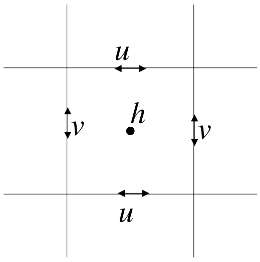
- Some older ocean models use B-grid. Eg ocean component of HadCM3.
- Gradients are compact but averaging needed
- Still computational modes
- Acurate representation of inertial oscillations (u and v are together)

C-grid



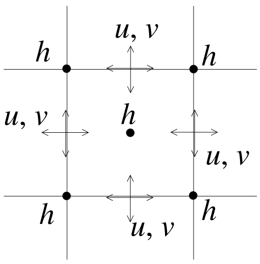
- Compact discretisation of $\partial u/\partial x$, $\partial u/\partial y$, $\partial h/\partial x$ and $\partial h/\partial y$ $\rightarrow$ best representation of gravity modes.
- Coriolis terms and geostrophic balance more of a problem.
- Widely used including current UK Met Office model.

D-grid



- Non-compact discretisation of $\partial u/\partial x$ and $\partial h/\partial x$ leads to spurious gravity modes
- Lots of averaging needed
- Best representation of geostrophic balance
- GFDL's FV3 model uses a C-D grid

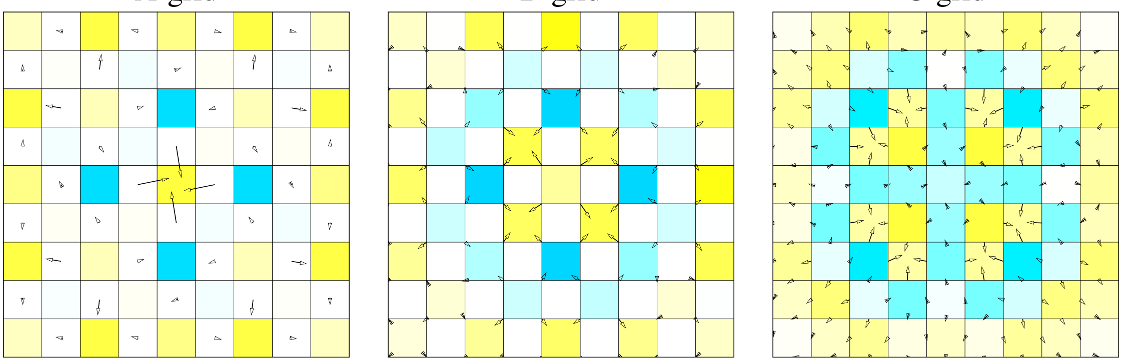
E-grid



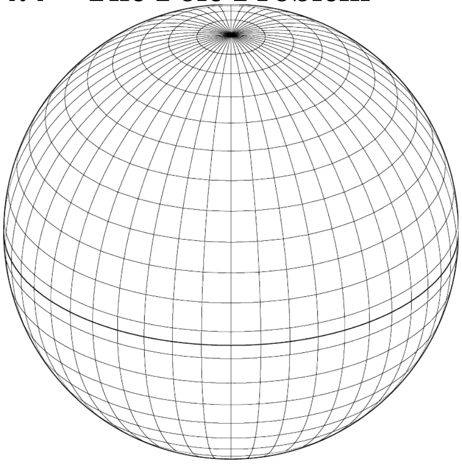
- Equivalent to a B-grid rotated by 45^o

4.3.1 Solutions of Linearised SWE starting from an initial bump

A-grid B-grid C-grid



4.4 The Pole Problem



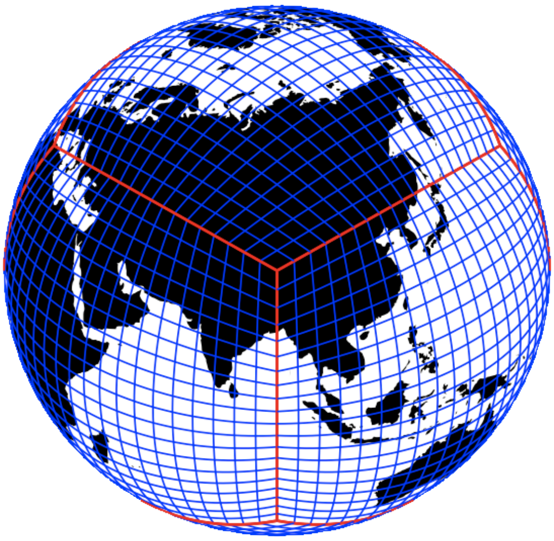
Convergence of points towards the poles

- Severe time-step restrictions
- Parallel scaling bottlenecks

Who uses a lat-lon grid?

- UK Met Office (moving to cubed sphere)
- NOAA (moving to cubed sphere)
- Environment Canada (for low resolution. Yin-Yang for high resolution)
- NCAR Community Atmosphere Model CAM-FV (for low resolution. CAM-SE on cubed sphere for high resolution)

4.4.1 Cubed Sphere

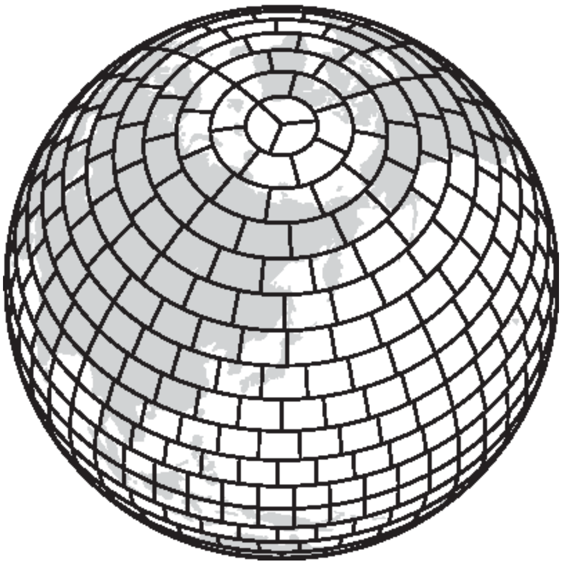


Grid lines are non-orthogonal which must be treated accurately to avoid grid imprinting.

Used by

- Next UK Met Office model
  - Mixed finite elements
- GFDL's FV3 (will also be used by NOAA)
  - finite volume, CD grid
- NCAR CAM-SE
  - spectral element method
- NUMA-NEPTUNE, from the US Navy (not operatation)
  - discontinuous Galerkin

4.4.2 Reduced grid

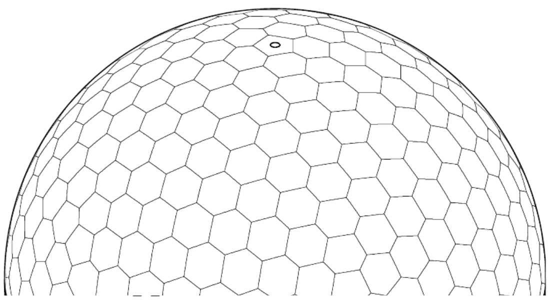


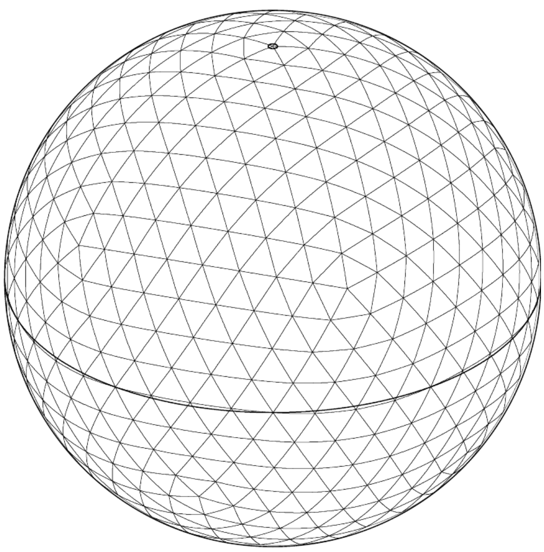
Used by

- ECMWF IFS
  - spectral method
- ECMWF FV3
  - experimental finite volume model using A-grid

4.4.3 Icosahedral grids

Suitable for low order models

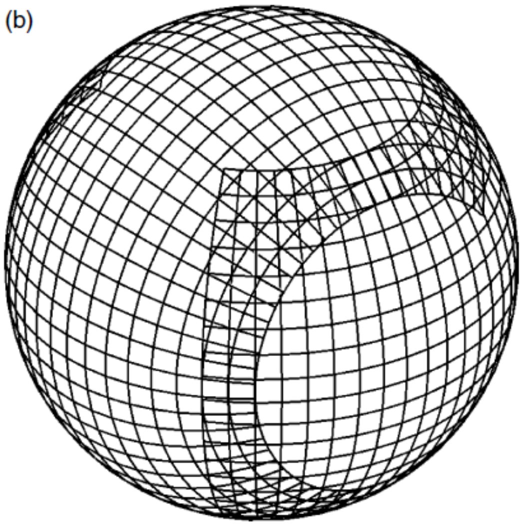




Used by

- NICAM
  - Japanese A-grid model run at very high resolution
- MPAS (Voronoi, US community model to replace WRF)
  - finite volume C-grid
- ICON (triangles, German operational model)
  - finite volume C-grid
- DYNAMICO (French experimental model)
  - finite volume C-grid
- Colorado State University Model (Uses Vorticity is a prognostic variable, Z-grid)

4.4.4 Yin-Yang Grid



Conservation is problematic
Spurious behaviour on overlaps

Used by

- Environment Canada
  - for high resolution modelling
  - semi-implicit, semi-Lagrangian (like UK Met Office)

4.5 Discretisations for terms of the Euler Equations

Linear Advection Non-linear advection
Terms involved in gravity and acoustic wave propagation
Coriolis

Momentum $$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u} = -2\mathbf{\Omega}\times\mathbf{u} - \frac{\nabla p}{\rho} + \mathbf{g}$$

Continuity $$\frac{\partial \rho}{\partial t} + \nabla\cdot\rho\mathbf{u} = 0$$

Potential temperature $$\frac{\partial \theta}{\partial t} + \mathbf{u}\cdot\nabla\theta = Q$$

An equation of state, eg perfect gas law, $p = \rho RT$

$\mathbf{u}$	Wind vector	$\mathbf{g}$	Gravity vector (downwards)
t	Time	θ	Potential temperature, $T\,(p_0/p)^{\kappa}$
$\mathbf{\Omega}$	Rotation rate of planet	κ	heat capacity ratio ≈ 1.4
ρ	Density of air	Q	Source of heat
p	Atmospheric pressure		

4.6 Who uses what for Advection?

Model/Modelling centre	Version	Numerical method
UK Met Office	Current	Semi-Lagrangian (not conservative)
	Next	Flux form semi-Lagrangian? (FV)
ECMWF	Current	Semi-Lagrangian (not conservative)
	Next	Flux form semi-Lagrangian (FV)
Environment Canada		Semi-Lagrangian (not conservative)
NOAA		Flux form semi-Lagrangian (FV)
NCAR CAM	FV	Flux form semi-Lagrangian (FV)
	SE	Spectral Element (no upwinding)
FV3 (NOAA and GFDL)		Flux form semi-Lagrangian (FV)
NUMA-Neptune (US Navy)	Next	Discontinuous Galerkin
NICAM (A Japanese model)		Flux form semi-Lagrangian (FV)
MPAS (to replace WRF)		Upwinded finite volume
ICON		Flux form semi-Lagrangian (FV)
DYNAMICO		Flux form semi-Lagrangian (FV)
CSU		Upwinded finite volume

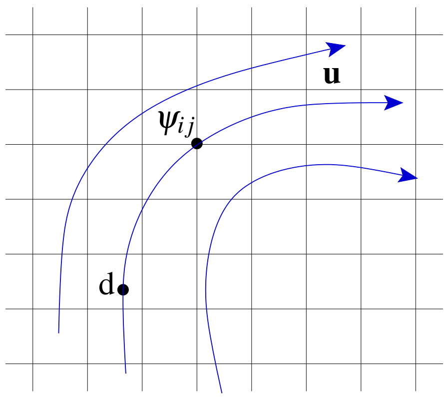
4.6.1 Semi-Lagrangian

$$\frac{\partial \psi}{\partial t} + \mathbf{u}\cdot\nabla\psi = 0$$

solved as

$$\psi_{ij}^{n+1} = \psi_d^n$$

where n is the time level, ij is the position on the grid and d is the departure point.
Interpolate to find ψ_d at the departure point from surrounding values on the grid.



Advantages and Disadvantages

☺ Stable and accurate for very long time steps

☺ Cost and accuracy not strongly related to time step

☹ ψ is not conserved

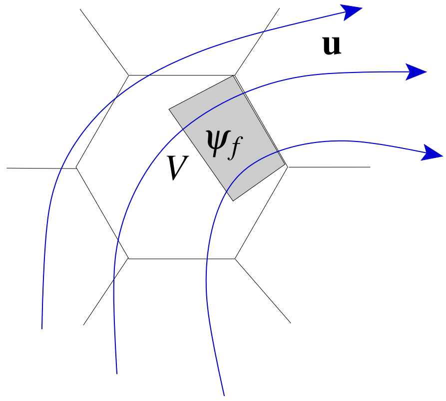
4.6.2 Flux Form Semi-Lagrangian

Also known as:

- Forward in time
- Swept area/volume
- Space-time

Examples:

- PPM (piecewise parabolic method)
- Lin and Rood
- COSMIC
- Lax-Wendroff.



☺ Conservative

☹ Only used with small time-steps ($c < 1$)

$$\frac{\partial \psi}{\partial t} + \nabla \cdot \psi \mathbf{u} = 0 \qquad \text{solved as} \qquad \psi^{n+1} = \psi^n - \frac{1}{V} \sum_{\text{faces}} \psi_f \mathbf{u} \cdot \mathbf{S}$$

where ψ_f is integrated over the volume swept through the face during one time step.

4.6.3 Other Advection Schemes

Space and time discretised separately – "method of lines".

Options for Space

- Finite difference (with upwinding)
- Finite volume (with upwinding)
- Spectral Element (not upwinded)
- Discontinuous Galerkin
- Finite Element (Petrov-Galerkin with upwinding)

Options for Time

- Runge-Kutta (multi-stage)
- Multi-step (eg leapfrog)
- Implicit (eg Crank-Nicolson)

4.7 Numerical Methods for Gravity and Acoustic waves (2nd order wave equations)

- Finite Difference - Arakawa A, B, C, D or E grids [Arakawa and Lamb, 1977]
  - Finite Volume (A-grid – co-located or C-grid – staggered)
- Spectral element (co-located)
- Discontinuous Galerkin (co-located)
- Spectral method (co-located)
- Finite Element
  - Mixed finite element – finite element version of staggered

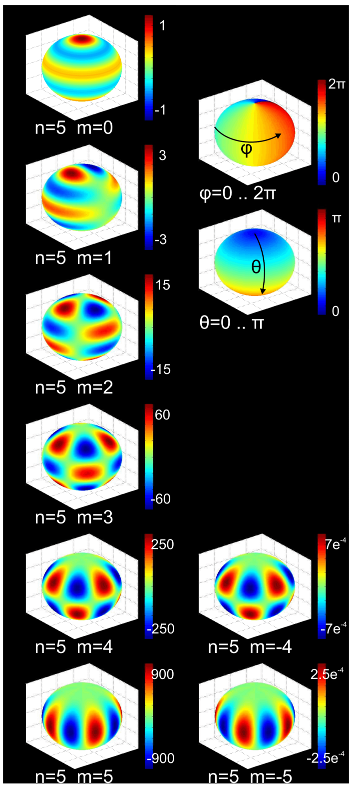
4.8 Spectral Method

- Fourier and Legendre transforms convert grid point data, $\tilde{X}(\lambda,\theta)$ (λ is longitude and θ is latitude) into coefficients, X_n^m so that the atmospheric state can be represented as a sum of spherical harmonics:

$$\tilde{X}(\lambda,\theta)=\sum_{n=0}^{N}\sum_{m=-n}^{n}X_n^m P_n^m(\theta)e^{im\lambda}$$

  where P_n^m is a Legendre polynomial.
- Differentiation and interpolation can then be done very accurately in spectral space.

Spherical harmonics for $n=5$ $\rightarrow$



Spectral Method

☺ Highest possible order of accuracy for the resolution

☺ ECMWF IFS has always been the most accurate model.

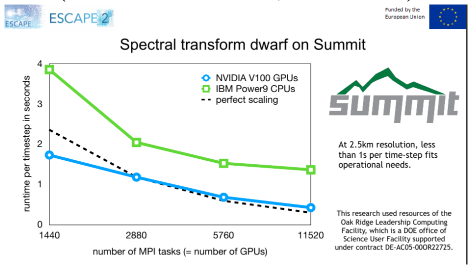
☺ No grid imprinting due to gird irregularities

☹ Lot of communication involved in spectral transforms and inverse transforms → parallel scaling problems

☹ Spectral ringing

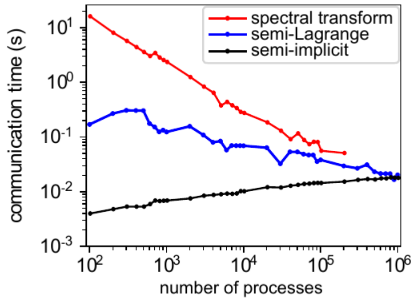
Scaling of spectral transform

(Andreas Müeller, ECMWF)



Performance modelling

[Zheng and Marguinaud, 2018]



4.9 The Finite Element Method

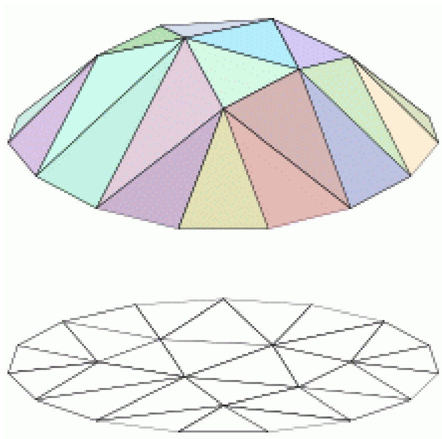
The solution, $u(\mathbf{x})$, is represented as a sum of N basis functions:

$$u(\mathbf{x}) = \sum_{i=1}^{N} U_i \phi_i(\mathbf{x}). \tag{4.6}$$

For the finite element method, the basis functions, ϕ_i, are piecewise polynomials defined as non-zero on on each element.

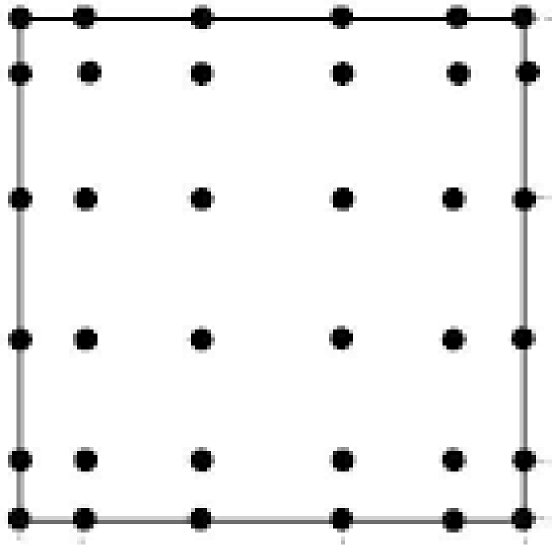
- To find U_i, multiply eqn (4.6) by a test function.
- Galerkin method:
  - test functions = basis functions
- Integrate by parts to get weak formulation
- Leads to a set of linear simultaneous equations: the mass matrix (or stiffness matrix) – **a global matrix**
- Solve to find U_i

Representation of a function using a piecewise linear basis.



4.10 Spectral Element Method

Basis functions are Lagrange interpolation on Gauss-Lobatto quadrature points

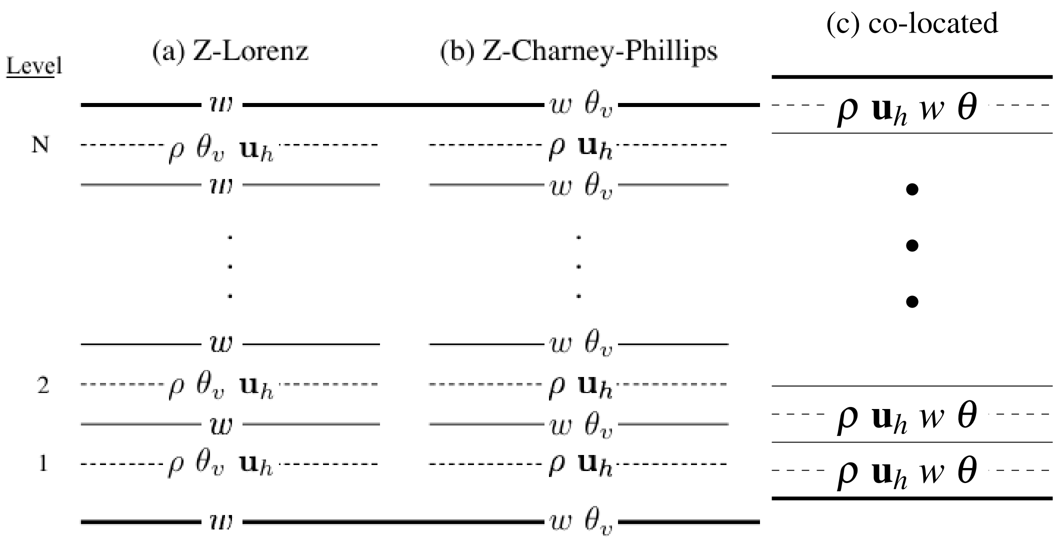


- Integration over an element is summation of values at Gauss-Lobatto quadrature points
- Leads to a diagonal mass matrix
  ∴ less communication
- Used by CAM-SE on the cubed sphere (4th order accurate)
- Excellent parallel scaling
- Pressure and velocity co-located
  ∴ A-grid computational mode but high order

4.11 Mixed Finite Element

- Different basis functions for pressure and velocity
  ∴ good wave dispersion
- Finite element equivalent of staggered grid
- Next generation Met Office model

4.12 Vertical Discretisation



UK Met Office
Environment Canada

MPAS, DYNAMICO
ICON, NICAM, CAM-SE

ECMWF IFS and FVM
GFDL and NOAA's FV^3
NUMA-Neptune, CAM-FV?

Bibliography

A. Arakawa and V. Lamb. Computational design of the basic dynamical processes of the UCLA general circulation model. *Methods in Computational Physics*, 17:173–265, 1977.

P. Colella and P.R. Woodward. The piecewise parabolic method (PPM) for gas-dynamical simulations. *J. Comput. Phys.*, 1984.

COMSOL: multiphysics cyclopedia. The finite element method (FEM). `https://uk.comsol.com/multiphysics/finite-element-method`, 2017. Accessed: 2019-06-18.

T. Dubos, S. Dubey, M. Tort, R. Mittal, Y. Meurdesoif, and F. Hourdin. DYNAMICO-1.0, an icosahedral hydrostatic dynamical core designed for consistency and versatility. *Geosci. Model Dev.*, 8(10):3131–3150, 2015.

ECMWF. *IFS Documentation CY45R1*, chapter Part III : Dynamics and numerical procedures. Number 3 in IFS Documentation. ECMWF, 2018. URL `https://www.ecmwf.int/node/18713`.

B.P. Leonard, A.P. Lock, and M.K. MacVean. Conservative explicit unrestricted-time-step multidimensional constancy-preserving advection schemes. *Mon. Wea. Rev.*, 124(11): 2585–2606, 1996.

S.J. Lin and R.B. Rood. Multidimensional flux-form semi-Lagrangian transport schemes. *Mon. Wea. Rev.*, 124:2046–2070, 1996.

R.B. Neale, J.H. Richer, A.J. Conley, S. Park, P.H. Lauritzen, A. Gettelman, and D.L. Williamson. Description of the NCAR Community Atmosphere Model (CAM 4.0).

Technical Report NCAR/TN-485+STR, NCAR, April 2010. URL `https://www.ccsm.ucar.edu/models/ccsm4.0/cam/docs/description/cam4_desc.pdf`.

M. Satoh, T. Matsuno, H. Tomita, H. Miura, T. Nasuno, and S. Iga. Nonhydrostatic icosahedral atmospheric model (NICAM) for global cloud resolving simulations. *J. Comput. Phys.*, 227(7):3486–3514, 2008.

W.C. Skamarock, J.B. Klemp, M.G. Duda, L.D. Fowler, S-H Park, and T.D. Ringler. A multi-scale nonhydrostatic atmospheric model using centroidal Voronoi tesselations and C-grid staggering. *Mon. Wea. Rev.*, 140(9):3090–3105, 2012.

P. A. Ullrich, C. Jablonowski, J. Kent, P. H. Lauritzen, R. Nair, K. A. Reed, C. M. Zarzycki, D. M. Hall, D. Dazlich, R. Heikes, C. Konor, D. Randall, T. Dubos, Y. Meurdesoif, X. Chen, L. Harris, C. Kühnlein, V. Lee, A. Qaddouri, C. Girard, M. Giorgetta, D. Reinert, J. Klemp, S.-H. Park, W. Skamarock, H. Miura, T. Ohno, R. Yoshida, R. Walko, A. Reinecke, and K. Viner. DCMIP2016: a review of non-hydrostatic dynamical core design and intercomparison of participating models. *Geosci. Model Dev.*, 10(12):4477–4509, 2017.

G. Zängl, D. Reinert, P. Rìpodas, and M. Baldauf. The ICON (ICOsahedral Non-hydrostatic) modelling framework of DWD and MPI-M: Description of the non-hydrostatic dynamical core. *Quart. J. Roy. Meteor. Soc.*, 141(687):563–579, 2015.

Y. Zheng and P. Marguinaud. Simulation of the performance and scalability of message passing interface (MPI) communications of atmospheric models running on exascale supercomputers. *Geosci. Model Dev.*, 11(8):3409–3426, 2018.