Numerical modelling of the atmosphere - 1

We have seen various processes and phenomena taking place in the atmosphere, on different spatial and temporal scales.

The coming 2 session focuses on **the numerical modelling of those processes** in order to
- predict weather and climate phenomena;
- Interpolate/complement observations.

Main objective: explain key concepts.

We will cover today:
1. The governing equations
2. The numerical solutions

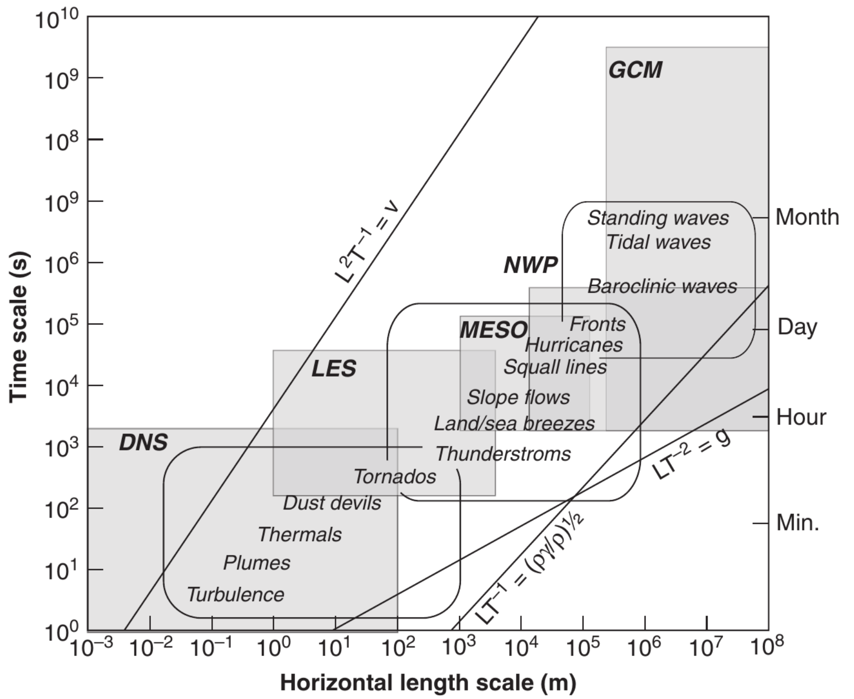
Book:
Warner, “Numerical weather and climate prediction”, 2011 → *W2011*

What is a numerical model?

A model is an abstract analogue of the actual phenomena (occurring in the atm)

A numerical model is the numerical implementation of an ensemble of equations assumed to represent the way actual physical variables are behaving.

There are different types of models depending on the atmospheric processes of interest and their typical spatial and temporal scales



Steyn, 2015

Conservation equations

Any atm model is based on a set of conservation principles (all relevant proc. are considered):

Mass: no source or sink, overall mass is constant.

Heat: atm supposed to be in thermodynamic equilibrium (~ideal gas)

Motion: Newton's 2nd law ($\Sigma\vec{F} = m\vec{a}$) with forces: Coriolis + pressure grad. + gravity + friction

Water: keep track of phase changes and mass fluxes (with source/sink terms)

Gaseous and aerosol material: same as water but for gaseous and aerosol matter...

Those equations are called the primitive equations.

Conservation equations

Quantity f in a fluid parcel advected with velocity $\vec{V} = (u, v, w)$

Advection operator for a scalar $\left(\vec{V}.\vec{\nabla}\right) f = u\frac{\partial f}{\partial x} + v\frac{\partial f}{\partial y} + w\frac{\partial f}{\partial z}$

Advection operator for a vector $$\left(\vec{V}.\vec{\nabla}\right) \vec{f} = \left[\begin{array}{c} (\vec{V}.\vec{\nabla})f_x \\ (\vec{V}.\vec{\nabla})f_y \\ (\vec{V}.\vec{\nabla})f_z \end{array} \right] = \left[\begin{array}{c} u\frac{\partial f_x}{\partial x} + v\frac{\partial f_x}{\partial y} + w\frac{\partial f_x}{\partial z} \\ u\frac{\partial f_y}{\partial x} + v\frac{\partial f_y}{\partial y} + w\frac{\partial f_y}{\partial z} \\ u\frac{\partial f_z}{\partial x} + v\frac{\partial f_z}{\partial y} + w\frac{\partial f_z}{\partial z} \end{array} \right]$$

Total or material derivative: $$\frac{Df}{Dt} = \frac{\partial f}{\partial t} + (\vec{V}.\vec{\nabla})f$$

Conservation equations

In mathematical (vector) terms

Mass: $$\frac{\partial \rho}{\partial t} = -\nabla(\rho \vec{V})$$

Heat: $$\frac{\partial \theta}{\partial t} = -\vec{V}\nabla\theta + S_\theta$$

Motion: $$\frac{\partial \vec{V}}{\partial t} = -\vec{V}\nabla\vec{V} - \frac{1}{\rho}\nabla P - g\vec{k} - 2\vec{\Omega} \times \vec{V} + \vec{F}_f$$

Water: $$\frac{\partial q_n}{\partial t} = -\vec{V}\nabla q_n + S_{q_n} \ , n = 1, 2, 3$$

Gas/aerosols: $$\frac{\partial \chi_n}{\partial t} = -\vec{V}\nabla\chi_n + S_{\chi_n} \ , n = 1..M$$

ρ	density	[kg m^{-3}]
$\vec{V}$	velocity	[m s^{-1}]
θ	pot. temp	[kg m^{-3}]
S_θ	source/sink heat	[K s^{-1}]
P	pressure	[Pa]
$\vec{k}$	vector from Earth center	[-]
$\vec{\Omega}$	angular vel.	[s^{-1}]
$\vec{F}_f$	friction force / mass	[N kg^{-3}]
q_n	mixing ratio water	[kg kg^{-1}]
S_{q_n}	source/sink water	[s^{-1}]
χ_n	mixing ratio gas/aerosol	[kg kg^{-1}]
S_{χ_n}	source/sink gas/aerosol	[s^{-1}]

Approximations

In the equations seen before, 3 characteristic propagation velocities:
sound velocity (acoustic waves), gravity wave velocity and wind velocity

To properly resolve those features, time step must smaller than dx/u (see section 5.2)

If fast wave phenomena are not relevant for our problem (ex: sound waves), we can simplify the equations.

Hydrostatic approximation: fluid at rest and pressure gradient = gravity.

$$\frac{\partial P}{\partial z} = -\rho g$$

In hydrostatic atm, no sound waves…

Valid for $\left|\frac{dw}{dt}\right| << g$, usually on horizontal scales > 10 km

Boussinesq approximation

The variations in density are neglected except for buoyancy:

$$\rho = \rho_0 + \rho' \quad \text{with} \quad \frac{\rho'}{\rho_0} << 1$$

If so, the mass conservation equation leads to $\vec{\nabla}.\vec{V} = 0$

Anelastic approximation

The elasticity of the air is neglected, but air still compressible.

$$\frac{\partial \rho}{\partial t} = -\nabla(\rho \vec{V}) \Rightarrow \frac{\partial \rho}{\partial t} = -\nabla(\rho_0 \vec{V})$$

Shallow fluid (or water) equations

Horizontal length scale is much greater than the vertical length scale
+ homogeneous, incompressible, hydrostatic fluid.

Mass conservation → vertical velocity scale << horizontal velocity scale.

Motion conservation → ver. pressure gradients ~ hydrostatic
hor. pressure gradients due to displacement of pressure surface

Hor. velocity is constant in the vertical and integrating over the atm column remove ver. vel.

Questions

1. What is a numerical model?

2. What are the conservation equations?

3. What is the hydrostatic approximation?

Primitive equations must be solved numerically

No analytical solution to those PDEs → numerical solutions. To do so, we need:

- Discretization of the domain of interest (global / regional for atm).
- Approximation of functions, derivatives, values…
- Initial + boundary conditions

We will see:

1. Numerical methods
2. Spatial grids
3. Vertical coordinates
4. Focus on Finite-difference method
5. Effects of numerical approximations

1. Numerical methods

4 main numerical frameworks to deal with the spatial dependence of the primitive equations

- **Finite-difference methods**: approximation of derivatives as finite differences.
  Pros: simple to implement, computationally less expensive than other approaches.
  Cons: not suitable for complex geometry, not always conservative.

- **Spectral methods**: Fourier or Legendre transform to obtain ODE instead of PDEs
  Pros: No non-linear instability, no spatial truncation error.
  Cons: spurious signal in the large gradients, computationally more expensive, non-conservative.

- **Finite-element methods**: local approximation over each element then globally optimized.
  Pros: can handle more complex geometry, usually more accurate.
  Cons: computationally more expensive.

- **Finite-volume methods**: integrated value over volume instead of grid point.
  Pros: conservative (mass, energy…) by design.
  Cons: computationally more expensive.

2. Spatial grids

The domain of interest must be discretized as a grid (Cartesian, lat-lon, spherical geodesic)

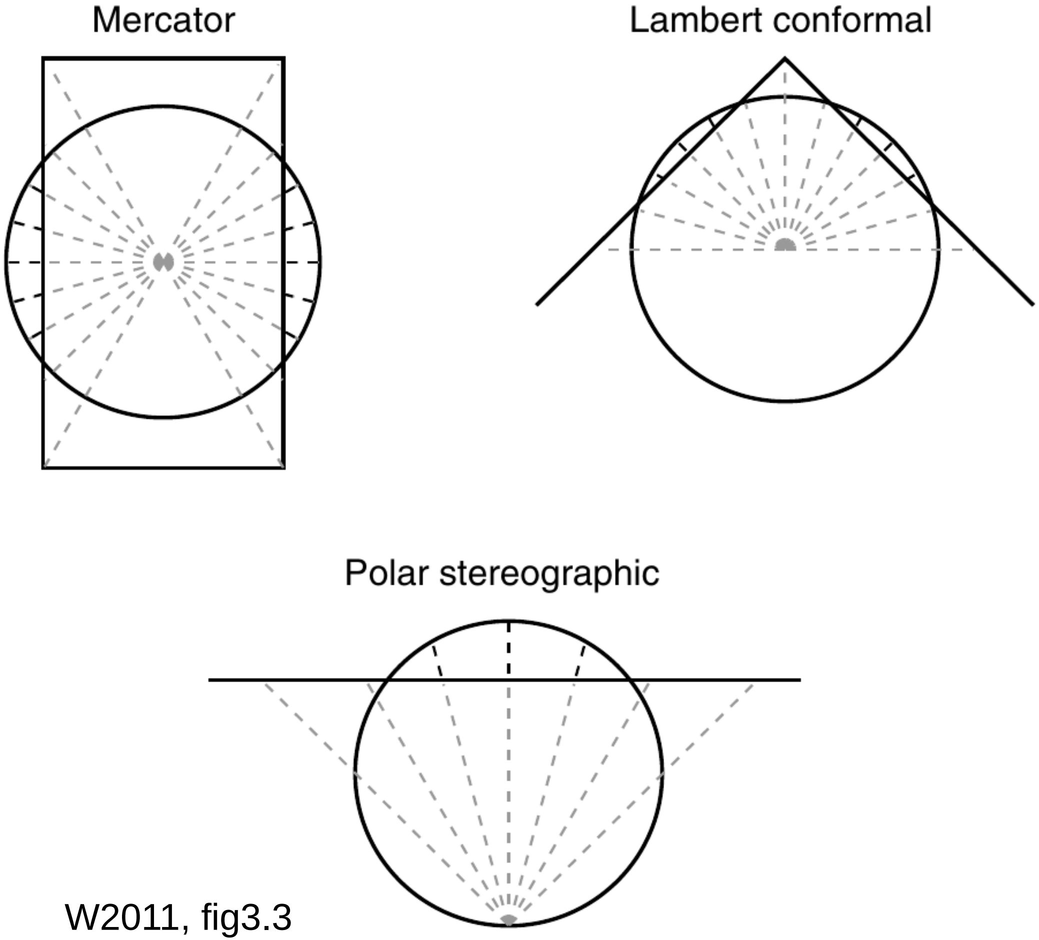
Grid increment (or spacing): time or space step in the grid.
Resolution: capability to distinguish parts or features of an object.
So resolution > grid increment...

Map projections: projection of 3D spherical coordinates on a 2D map. Ex: Mercator, Eckert, Lambert…

Conformal projections: the angle of two crossing curves is preserved (at local scales). Area is not preserved. Ex: Mercator, Lambert, stereographic.
Relevant for atm modelling (ex: wind direction) over regions with limited distorsion.

Map projection is important for limited-area (regional) models that work on Cartesian grids.

Examples of projections



W2011, fig3.3

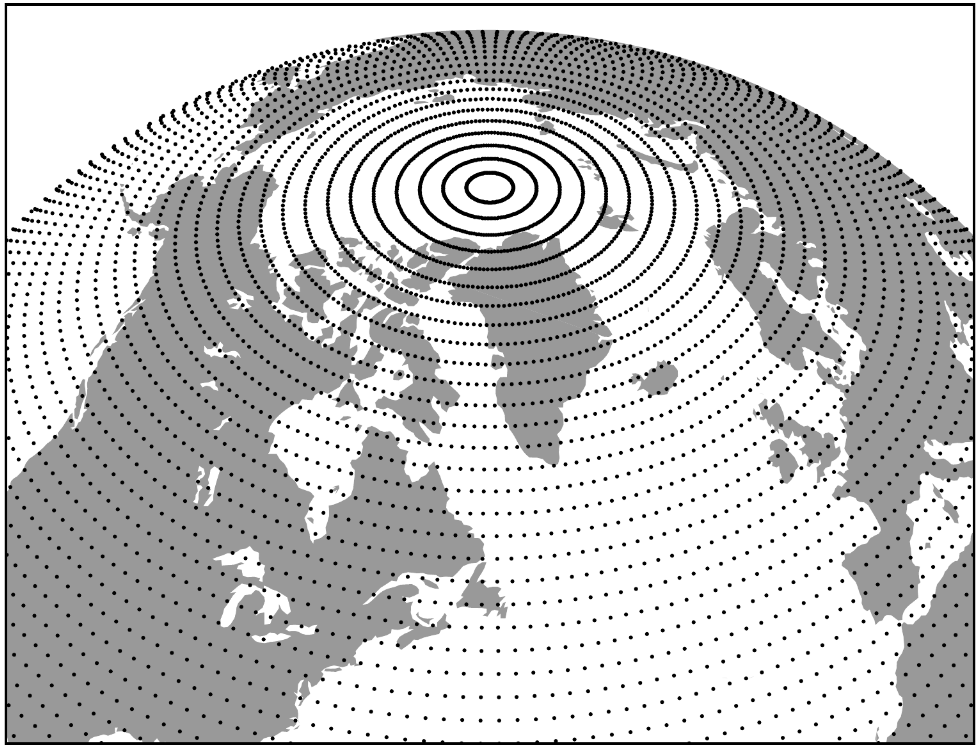
Latitude-longitude grids

Horizontal coordinates = latitude and longitude, vertical along radial from Earth's center.

Grid = increments in latitude and longitude.

Pros: no deformation.

Cons:
- Distances become smaller towards poles
  (requires short time steps → comp. cost)
- Singularities at the poles.



W2011, fig3.8

Latitude-longitude grids

Reduced grid: longitude increment is not constant (larger close to the poles)

Reduce the density of points near the poles so the computational cost.

But singularities still there...

W2011, fig3.18: reduced Gaussian T106 grid

Spherical geodesic grids

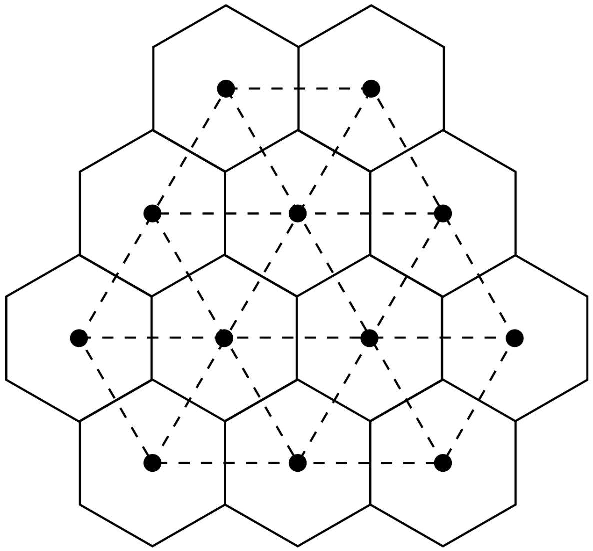
Geodesic = shortest distance between 2 points on a curved surface.

Spherical geodesic grid = ensemble of spherical, equilateral triangles (edges = geodesics)

Basic pixel shape = triangles or hexagons

Pros:
- Nearly homogeneous density of points over the sphere
- Easy to locally increase resolution (e.g. near mountains)

Cons:
- Pixel indexing is more complicated



W2011, fig3.12

Spherical geodesic grids

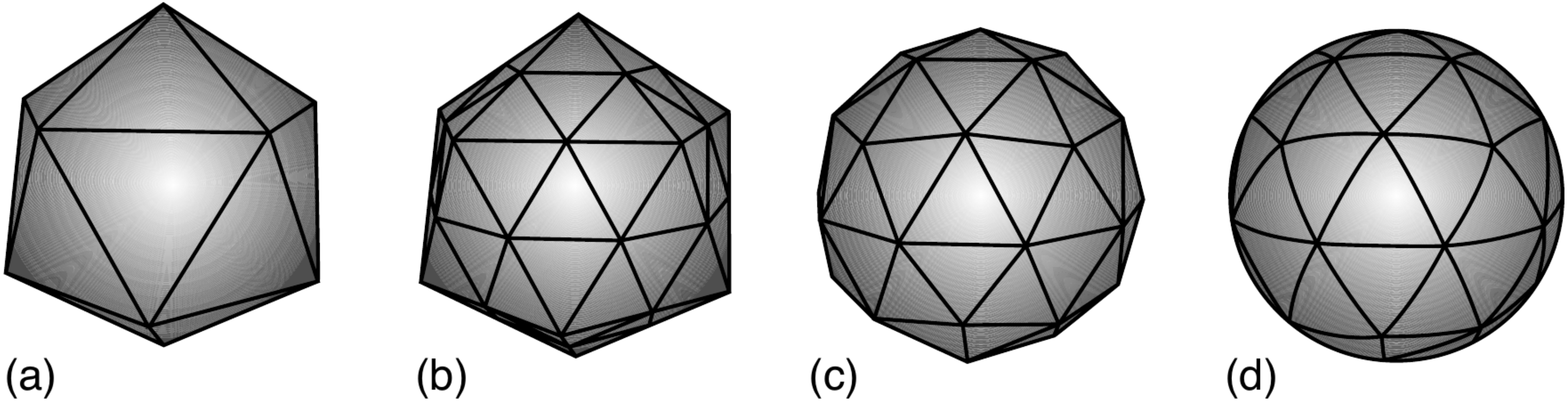


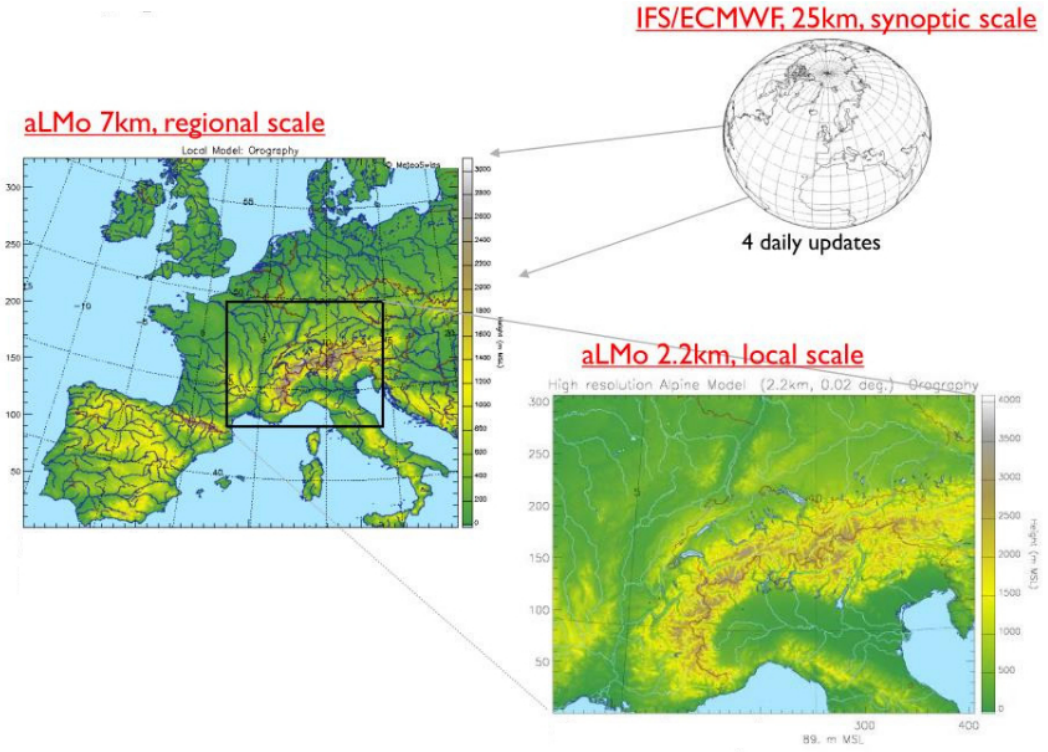
Fig. 3.10 In the generation of a spherical geodesic grid, the major triangles of the icosahedron (a) are subdivided, where (b) shows one approach. The vertices of the new triangles are projected (c) onto the sphere that is coincident with the vertices of the icosahedron. Geodesic lines are then drawn between the new vertices to generate spherical grid triangles (d).

W2011, fig3.10

Nesting grids

Applications focused on small-scale regions or local processes require higher horizontal resolutions that what can be achieved for a global model → domains/grids are nested

Ex: nested domains for COSMO at MeteoSwiss



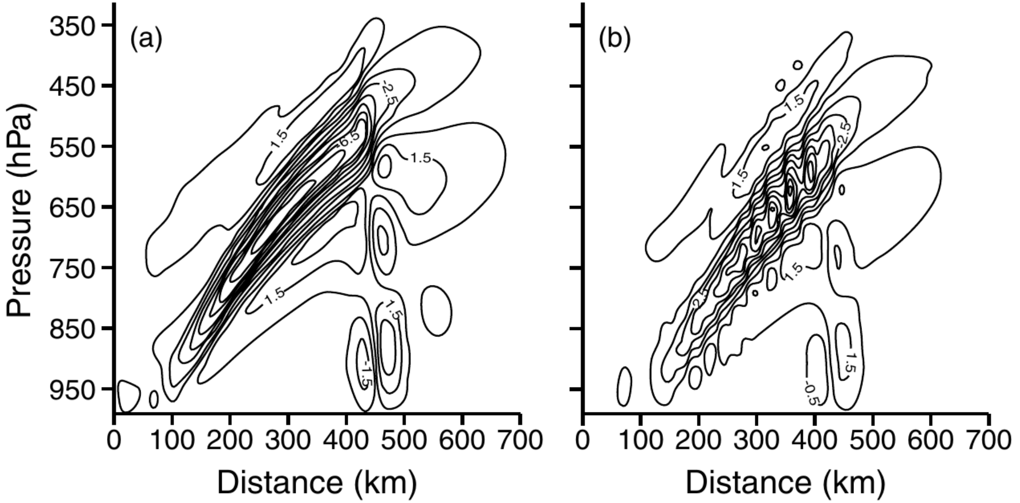
Consistency between vertical and horizontal increments

The physical features of atm phenomena must be properly resolved in the horizontal and the vertical to obtain realistic simulations.

→ horizontal and vertical increments cannot be independently set: $\Delta z_{opt} = s\Delta x$
where s = slope of the front (0.005 – 0.02).

This relationship is valid overall (order of magnitude).

If not properly set, spurious gravity waves may appear...



W2011, fig3.16 **Fig. 3.16** Vertical cross sections of vertical velocity, ω (solid lines, μb s^{-1}) after 24-h simulations of conditional symmetric instability with a 10-km horizontal grid increment and 75 layers (a) and 25 layers (b). From Persson and Warner (1991).

3. Vertical coordinates

1. Height above sea level

Seems most logical, but issue in complex terrain: altitude contours intersect terrain → undefined atm properties at those points…

2. Pressure level

Similar issues to height level, exacerbated because variability of pressure in time…

3. Potential temperature

In adiabatic atm, air parcel remains at same θ, and vertical movement related to change in θ.
Pros: const res. in θ implies better resolution where strong gradients of θ.
Cons: same as above in complex terrain + non-adiabatic regions (PBL, phase change)

2. Numerical solutions

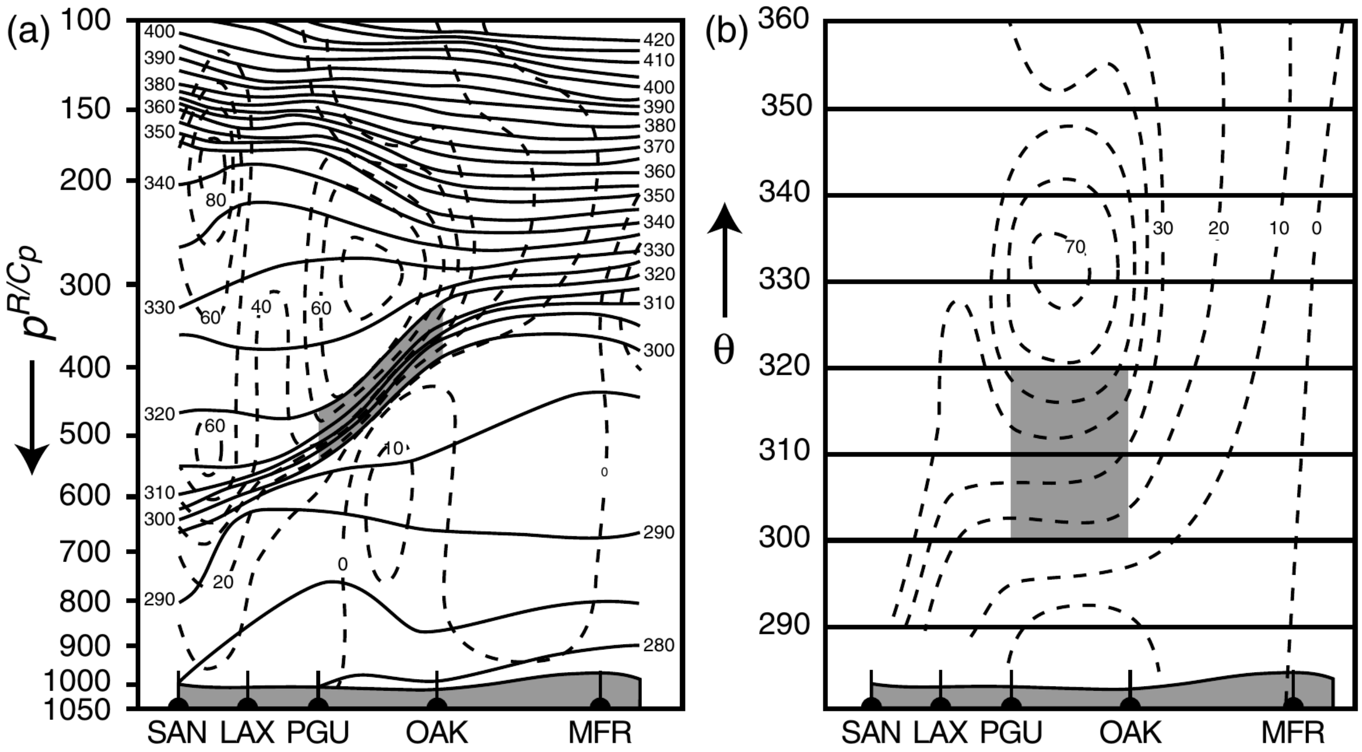


Fig. 3.37 Cross section of a front in pressure (a) and isentropic (b) coordinates, where the horizontal axis is north–south along the coast of the western USA. The gray area in the two cross sections spans the same volume of atmosphere. The solid lines are isentropes and the dashed lines are isotachs. From Benjamin (1989), based on Shapiro and Hastings (1973) and Bleck and Shapiro (1976).

W2011, fig3.37

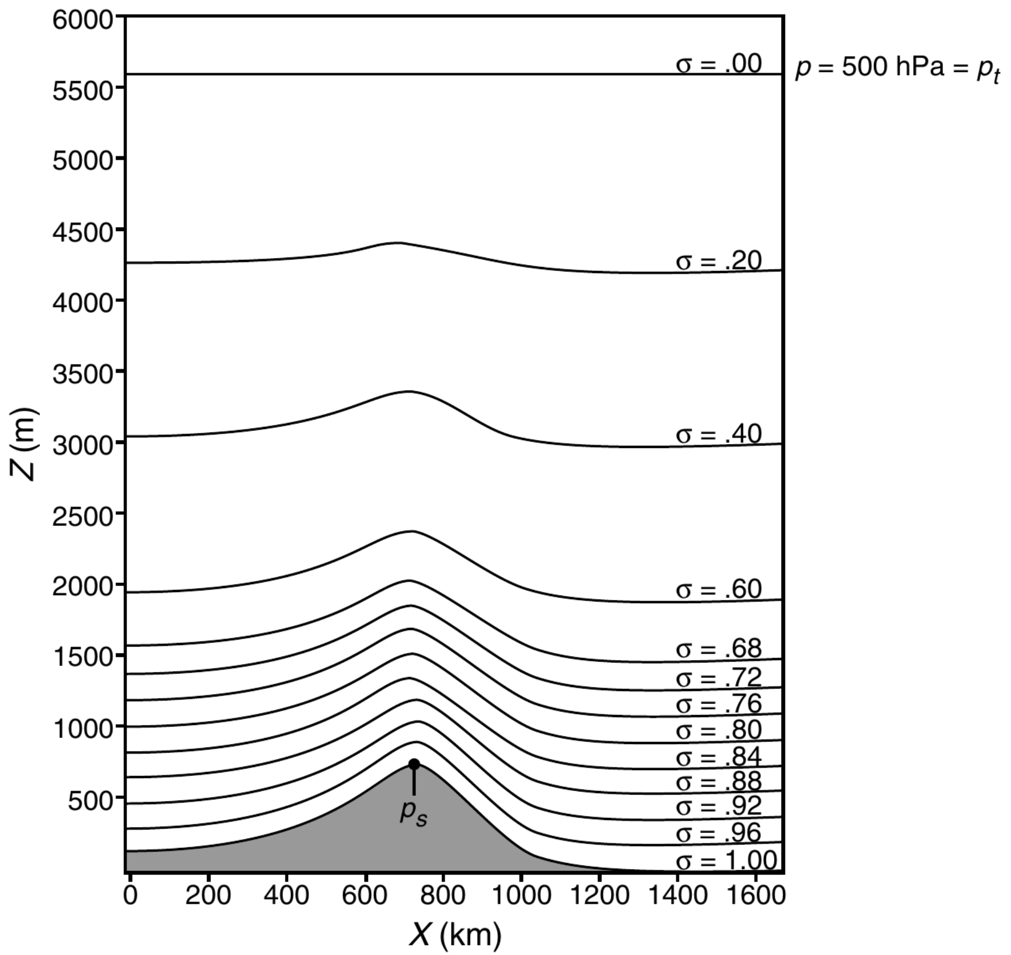
4. Sigma-p

Terrain-following coordinate: $$\sigma = \frac{P - P_t}{P_s - P_t}$$

P_t	constant pressure at top of domain
P_s	pressure at surface
P	pressure at a given location in domain

σ follows terrain and is in [0,1]
but varies in time (as P)

Sigma-z is similar but with altitude ref levels,
does not vary in time.



W2011, fig3.38

5. Hybrid isentropic-sigma

Terrain-following coordinate σ in the lower troposphere, isentropic coordinates above

η or step-mountain coordinates:

$$\eta = \eta_s \sigma = \frac{P_{ref}(z_s) - P_t}{P_{ref}(0) - P_t} \times \frac{P - P_t}{P_s - P_t}$$

P_t	constant pressure at top of domain
P_s	pressure at surface
P	pressure at a given location in domain
$P_{ref}(z)$	reference pressure as function of geometric height z
z_s	geometric height at interface between model layers

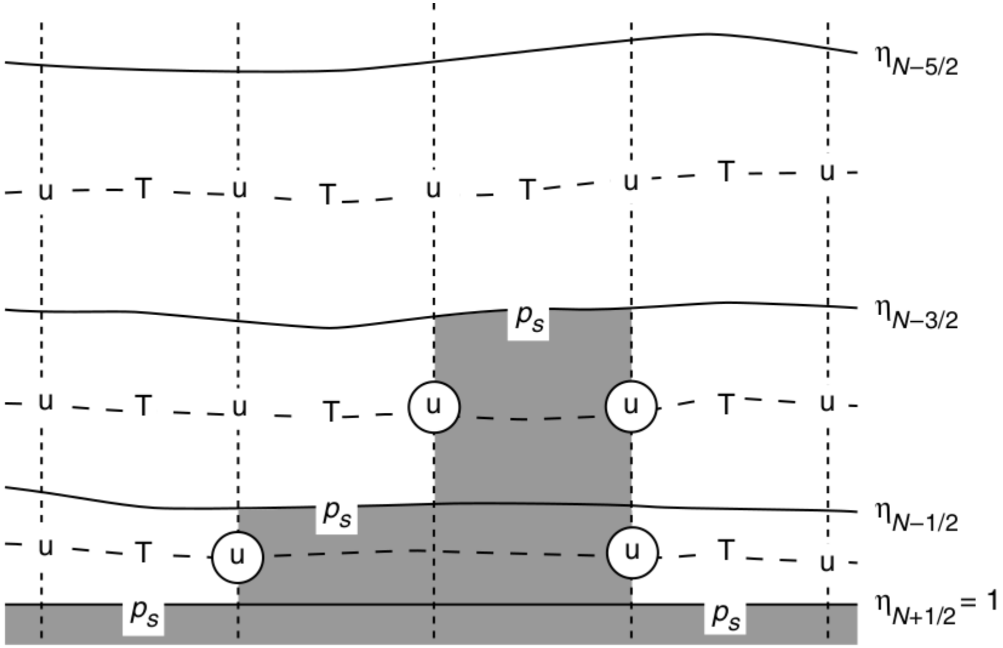


Fig. 3.39 Cross section of the three lower model levels for the step-mountain coordinate system, showing where variables are defined. The shaded area represents the land surface. From Mesinger *et al.* (1988).

W2011, fig3.39

Questions

1. Cite the 3 main types of spatial grids we have seen.

2. Can the vertical increment be set independently of the horizontal one?

3. What is the issue with pressure vertical coordinates in complex terrain?

4. Finite-difference methods

Recall that finite-difference methods are easy to implement and computationally cheap...

1. Time-differencing methods

Time-differencing methods can be explicit, implicit or a combination of both

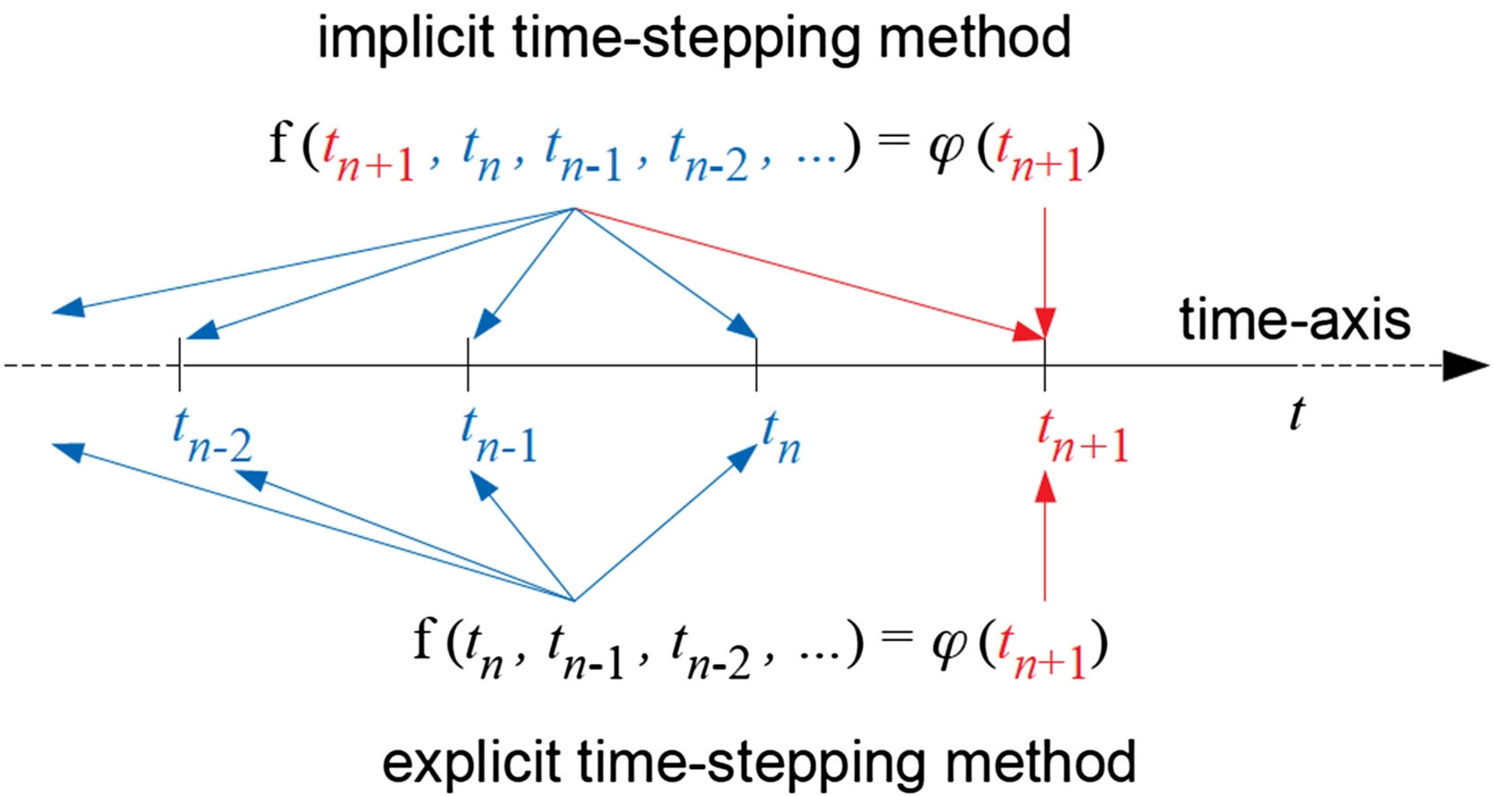
Explicit: left term of prognostic eq → variable at new time, right term → at previous time(s)

Implicit: variable at new time in both left and right terms → iterative solution

Semi-implicit: some terms are solved explicitly, some implicitly (ex: Krank-Nicolson method)

For a given variable ϕ:

Explicit scheme: $\dfrac{\partial \phi_i^{\tau+1}}{\partial t} \simeq \dfrac{\phi_i^{\tau} - \phi_i^{\tau-1}}{\Delta t}$ Implicit scheme: $\dfrac{\partial \phi_i^{\tau+1}}{\partial t} \simeq \dfrac{\phi_i^{\tau+2} - \phi_i^{\tau}}{2\Delta t}$



Sankaran, 2019

2. Space-differencing methods

Similarly to time-differencing, you can have explicit or implicit methods based on grid points.

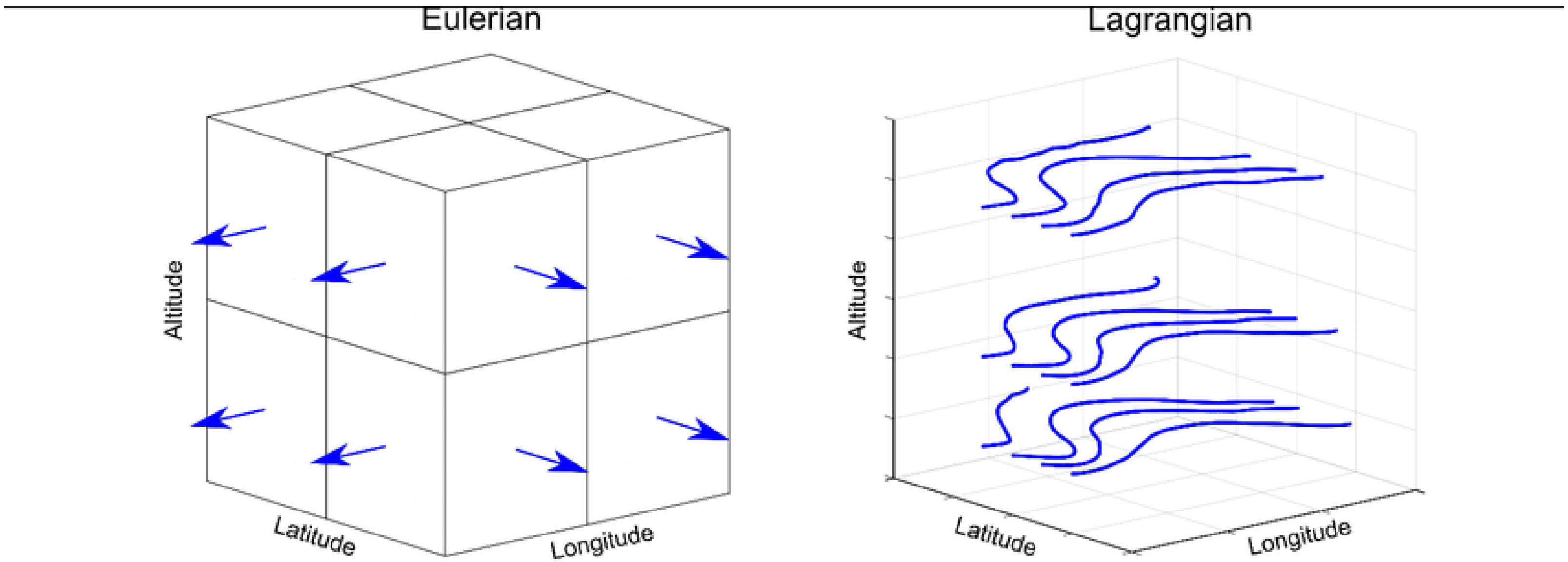
This is a Eulerian approach (fixed grid in space and time). Various schemes: forward-upstream, leapfrog, Adams-Bashford...

But this approach requires short time steps for numerical stability so large computational costs.

The Lagrangian approach follows the air parcel → total derivative.
Conserved quantity → pure advection

Semi-Lagrangian approach to address Lagrangian deficiencies: after a few iterations parcels would be very unevenly distributed in space (following winds). To do so, regularly spaced parcels are released at each time step. Not always conservative...

2. Numerical solutions

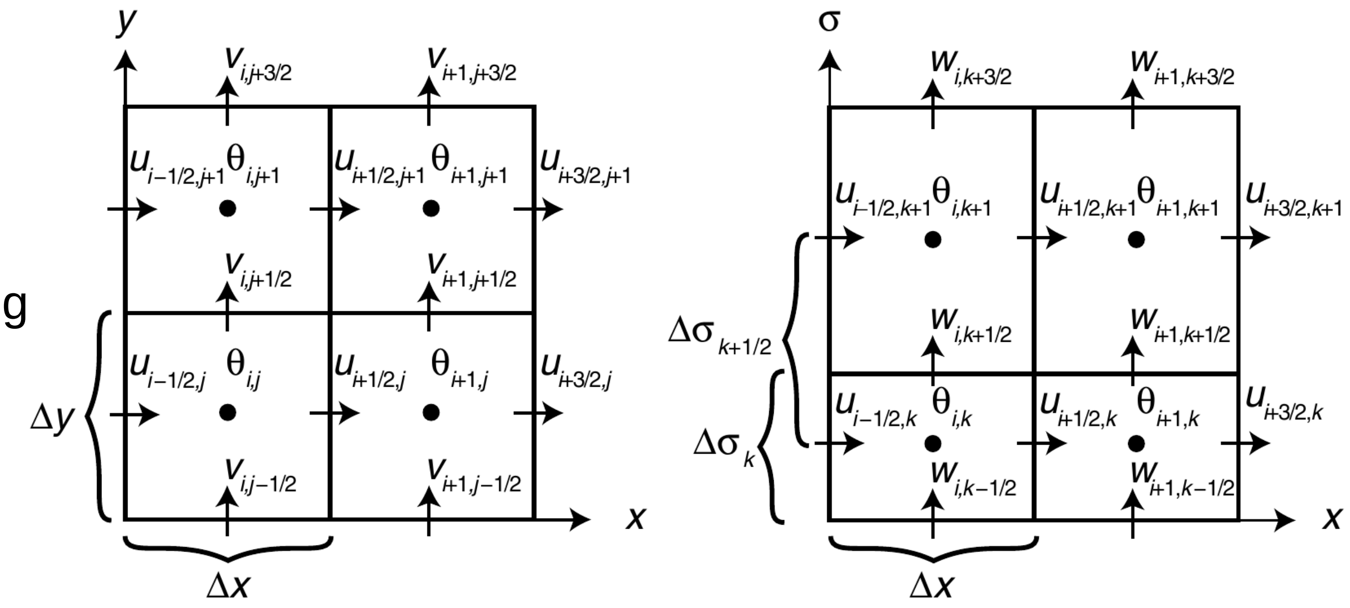


Tuinenburg HESS (2019)

3. Grid staggering

Different variables are defined on staggered grids (typically by 0.5 increment).
So derivatives can be estimated on smaller intervals.
→ Gain in resolution + decrease in truncation errors.

Arakawa-C grid-staggering



W2011, fig3.21 Horizontal grid Vertical grid

5. Effects of numerical approximations

All the listed numerical methods are based on approximations → errors in obtained solutions

1. Truncation error

Derivatives are approximated by truncated Taylor's series.

$$f(x) = f(a) + (x-a)\frac{\partial f}{\partial x}(a) + \frac{(x-a)^2}{2!}\frac{\partial^2 f}{\partial^2 x}(a) + \ldots + \frac{(x-a)^n}{n!}\frac{\partial^n f}{\partial^n x}(a) + R(n,x)$$

3-point differencing scheme: $f(a \pm \Delta x) = f(a) + (\pm \Delta x)\frac{\partial f}{\partial x}(a) + \ldots$

→ 2nd-order accuracy approx: $\frac{\partial f}{\partial x}(a) \simeq \frac{f(a+\Delta x) - f(a-\Delta x)}{2\Delta x}$

Example: sine function

Plot shows ratio between approx and true value, as a function of number of increments per wavelength.

Ratio > 0.9 for n ~ 8-10

So you need ~8-10 increments to properly capture the derivative of a wave...

W2011, fig3.22

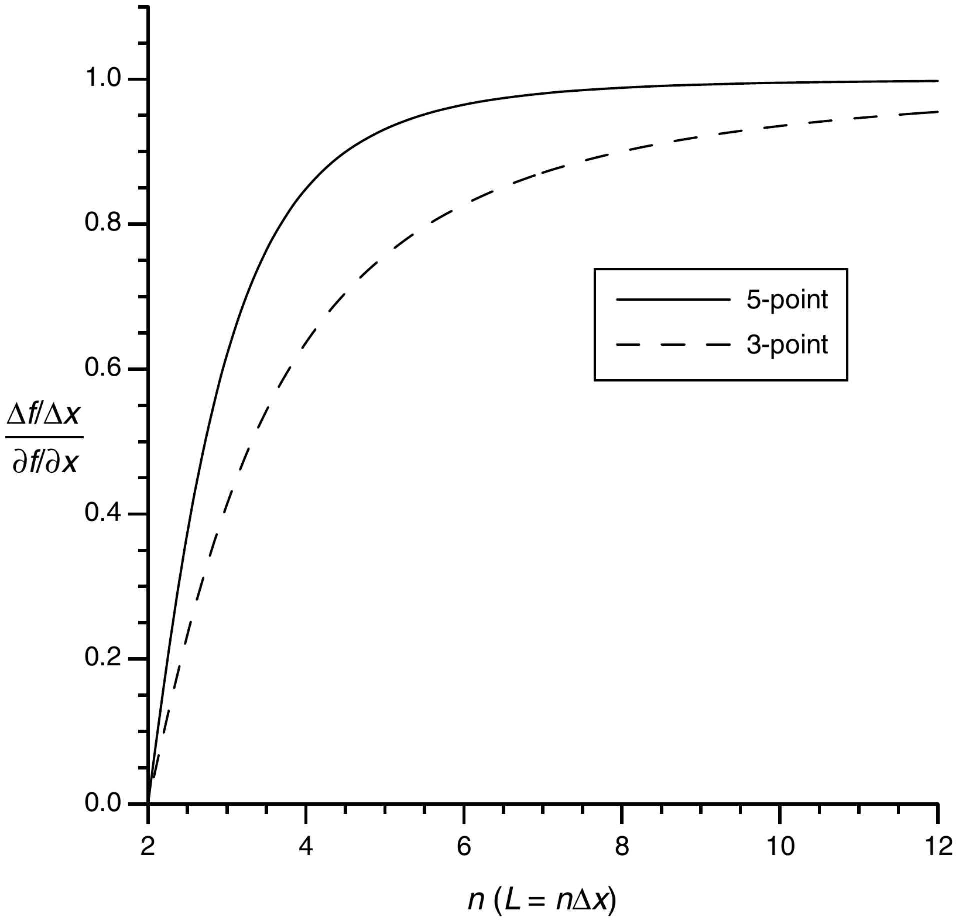


Fig. 3.22 The ratio of the value of the numerical approximation to the derivative of the cosine function and the value of the true derivative, for different numbers of grid increments per wavelength (how well the wave is resolved), for the five-point (fourth-order) and three-point (second-order) approximations.

2. Time integration

Stability of an (atm) numerical model: whether amplitudes of waves in (numerical) solutions grow exponentially because of non-physical reasons.

Advection terms in the conservation equations are the most problematic for stability.

Courant-Friedrichs-Lewy (CFL) linear stability condition: $C = U\frac{\Delta t}{\Delta x} < C_{max}$

C = Courant number and C_{max} ~1

→ The time step must be small enough so the fastest features are properly approximated.

If model allows sound waves, U ~ 300 ms^{-1} → strong constraint on the time step!

3. Diffusion

Diffusion: spatially spread features in heat, moisture and momentum fields.

Due to physical processes (turbulence) but also numerical methods (explicit, implicit, grid)*.

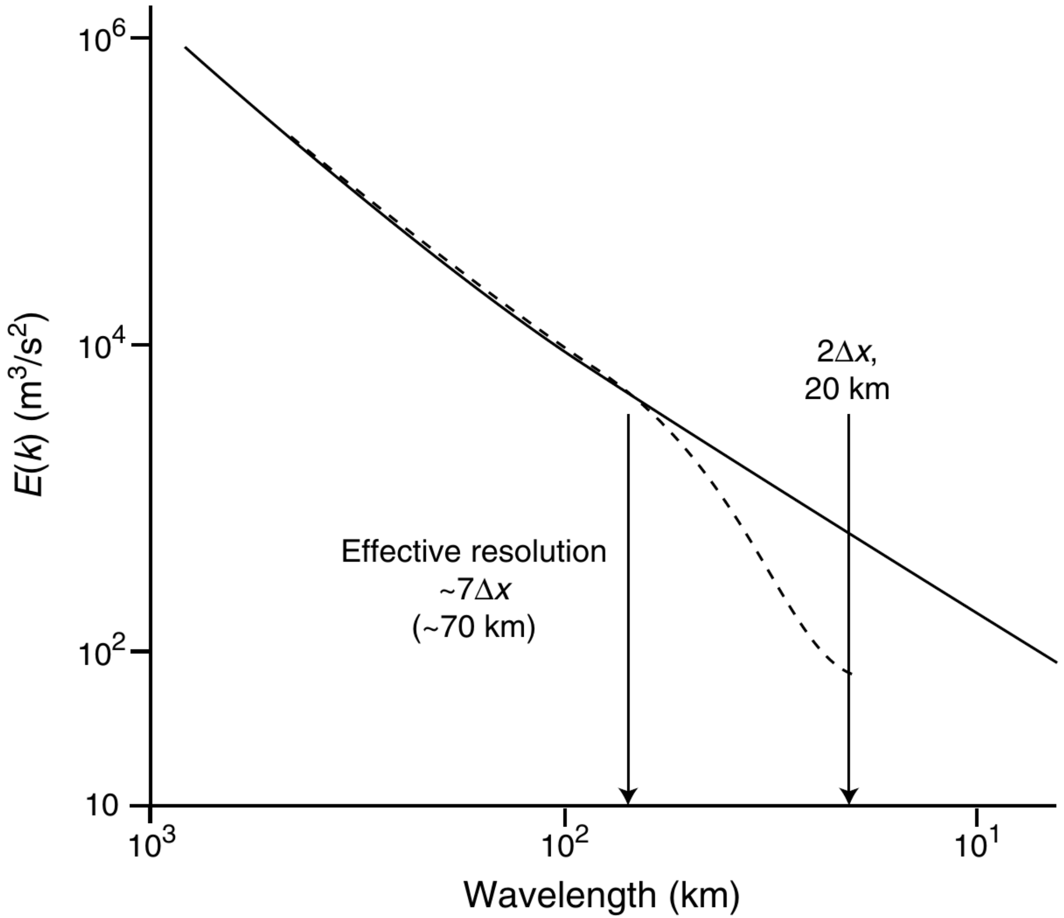
Explicit diffusion: introduced in models to clean up unrealistic features (due to boundary-conditions noise, computational scheme, energy aliasing).

Implicit diffusion: some numerical schemes selectively filter wavelength bands (ex. Runge-Kutta). If well controlled, may relax the need to add explicit diffusion.

Grid diffusion: derivative at a given grid point is influenced by neighboring grid points → diffusion of properties depending on grid increment and time step, for spatial-differencing schemes.

* terminology from W2011

2. Numerical solutions



W2011, fig3.36

Fig. 3.36 The effect of diffusion on the kinetic-energy spectrum for a WRF-model forecast having a 10-km grid increment. The expected slope of $k^{-5/3}$ is shown as a reference, and is reproduced by the model for wavelengths above $7\Delta x$. But the energy between the $2\Delta x$ and $7\Delta x$ wavelengths has been damped by the diffusion, resulting in an effective resolution of 70 km, not 20 km. Adapted from Skamarock (2004).

Questions

1. Explain the difference between a Eularian and a Lagrangian approach.

2. What are the truncation errors?

3. Why is there diffusion in numerical models?

Why all this?

Let’s have a look at the description of the dynamical core of the IFS model given by ECMWF:

“The dynamical core of IFS is hydrostatic, two-time-level, semi-implicit, semi-Lagrangian and applies spectral transforms between grid-point space (where the physical parametrizations and advection are calculated) and spectral space. In the vertical the model is discretised using a finite-element scheme. A reduced Gaussian grid is used in the horizontal.”

https://www.ecmwf.int/en/research/modelling-and-prediction/atmospheric-dynamics

Summary

Numerical modelling - 1

1. Governing equu.

→ Conservation equations (mass, heat, motion, water...)
→ Approximations (hydrostatic, Boussinesq, anelastic, shallow fluid)

2. Numerical sol.

→ Main types of numerical methods (finite-diff/element/volume, spectral)
→ Spatial grids (projections, lat-lon, geodesic)
→ Vertical coordinates (height, pressure, potential temp., sigma-p/z)
→ Finite-difference methods (time/space-differencing, grid staggering)
→ Effects of numerical approximations (truncation, time constraint, diff.)