ENDGame Dynamical Core

Thomas Melvin and Markus Gross

Met Office, Exeter, Devon

ENDGame...

- Even
- Newer
- Dynamics for
- General
- atmospheric
- modelling of the
- environment

...is the next dynamical core for the Unified Model (UM)

- single model for:
  - Weather forecasts ($25Km \rightarrow 1Km$, hours-days)
  - Climate simulations ($100Km$ 10-100 years)
  - Research tool $> 10m$

Design Philosophy

- Met Office philosophy: use unapproximated equations; use numerics to do "filtering"
- Fully compressible, nonhydrostatic models do not filter the acoustic modes
  - Have to be handled implicitly if wish to avoid severe restriction on time step
- Deep atmosphere models have twice as many Coriolis terms to handle
  - Larger stencil if terms handled implicitly which stability requires for two-time-level scheme
- But, more accurate; more general (eg planetary atmospheres)
- Do not want to introduce any computational modes
- Does not need diffusion/filtering for stability

Operational requirements (NWP)

Global 25km model (2010):

- Forecast to: 7 days 3 hours
- Timestep: = 10mins $\rightarrow$ 1026 time steps
- Resolution $N512L70 \rightarrow 1024 \times 768 \times 70 = 55$M grid points
- To run in 60 minute slot, including data assimilation and output

36 ($\approx 2^5$) times bigger than running 5 years ago

Features

- Evolution of the current New Dynamics dynamical core
- Uses a latitude-longitude grid
- Optional ellipsoidal geopotential approximation
- Switches to allow hydrostatic/shallow atmosphere approximations
- Improved handling of Coriolis terms on staggered grid
- (Almost) same grid as New Dynamics - C-grid + Charney Philips but with v-at-poles
- Fully implicit (iterative) scheme for all non-advective terms
- Consistent SL scheme for all variables (+SLICE option)
- Physics coupling:
  - Parallel split for slow processes, sequential and iterative for fast processes

Equations

$$\frac{D_r u}{Dt} - \frac{uv\tan\phi}{r} - 2\Omega\sin\phi v + \frac{c_{pd}\theta_{vd}}{r\cos\phi}\frac{\partial\Pi}{\partial\lambda} = -\left(\frac{uw}{r} + 2\Omega\cos\phi w\right) + S^u$$

$$\frac{D_r v}{Dt} - \frac{u^2\tan\phi}{r} + 2\Omega\sin\phi u + \frac{c_{pd}\theta_{vd}}{r\cos\phi}\frac{\partial\Pi}{\partial\phi} = -\left(\frac{vw}{r}\right) + S^v$$

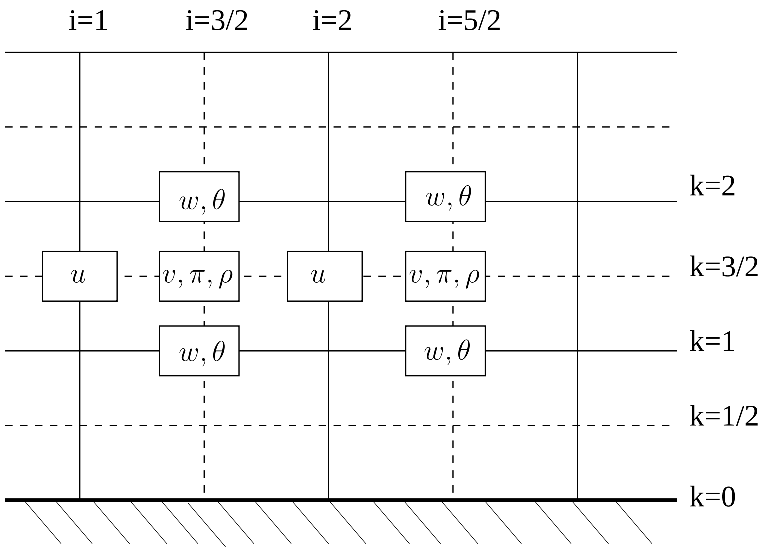
$$\frac{D_r w}{Dt} + c_{pd}\theta_{vd}\frac{\partial\Pi}{\partial r} + \frac{\partial\Phi}{\partial r} = \frac{\left(u^2 + v^2\right)}{r} + 2\Omega\cos\phi u + S^w$$

$$\frac{D_r}{Dt}\left(\rho_d r^2\cos\phi\right) + \rho_d r^2\cos\phi\left(\frac{\partial}{\partial\lambda}\left[\frac{u}{r\cos\phi}\right] + \frac{\partial}{\partial\phi}\left[\frac{v}{r}\right] + \frac{\partial w}{\partial r}\right) = 0$$

$$\frac{D_r\theta_{vd}}{Dt} = S^\theta$$

Spatial Discretisation

- C-grid horizontal staggering + Charney Philips vertical grid



- Uses height as vertical coordinate - terrain following at lower boundary
- Discretise equations using second order centred finite differences

$$\left(\frac{\partial F}{\partial x}\right)_{i-1/2} = \frac{F_i - F_{i-1}}{\Delta x}$$

Temporal Discretisation

- Two-time-level Semi-Implicit Semi-Lagrangian discretisation

$$\frac{Dq}{Dt} = F(q) \rightarrow q_A^{n+1} - q_D^n = \Delta t \overline{F(q)}^t$$

- Option to time offcentre terms

$$\overline{G}^t = \alpha G^{n+1} + (1-\alpha) G_D^n$$

Temporal Discretisation

- Use iterative approach to handle nonlinear terms

$$\frac{Dq}{Dt} = F(q) \rightarrow q_A^{(k+1)} - q_D^n = \alpha \Delta t \left[L(q)^{(k+1)} + (F(q) - L(q))^{(k)} \right]$$

$$+ (1-\alpha)\Delta t F(q)_D^n$$

- for centred scheme $(\alpha = 1/2)$ at convergence this reduces to a second order accurate Crank-Nicolson scheme
- in practice $\alpha > 1/2$ and scheme is not run to convergence (fixed number of iterations used)

Departure points

For semi-Lagrangian scheme need to solve trajectory equation $\frac{Dx}{Dt} = u$

- ENDGame uses local cartesian departure point scheme
- $x_D = x_A - \frac{\Delta t}{2}\left[u_A^{n+1} + u^n\left(x_D\right)\right]$
- This gives a doubly implicit equation (depends on $x_D,\ u^{n+1}$)
    - This is solved in an iterative manner

Iterative algorithm

Do time-step loop:

- given $(\dot{\eta}, \theta, w, u, v, \rho, \pi)^n$ at level n
- compute slow physics terms, (radiation, gwd etc.)

Do outer-loop iteration:

- compute SL departure points (x_D^n, y_D^n, z_D^n) using $(u, v, w)^n$ and latest estimate for $(u, v, w)^{n+1}$
- interpolate time level n terms to departure points for required fields
- compute predictors for timelevel $n + 1$ fields for use by fast physics terms
- compute fast physics increments, (convection, boundary layer etc.)

Do inner-loop iteration:

- evaluate Coriolis and nonlinear terms
- solve Helmholtz problem for π^{n+1}
- update estimate for prognostic variables at timelevel $n + 1$

Enddo

Enddo

Enddo

Conservation

- ENDGame is not inherently conservative:
  - Solve advective form of equations
  - SL interpolation is inherently dissapitive
  - Charney-Phillips grid staggers tracers wrt. mass
- However, option to use SLICE -
  - Semi-Lagrangian Inherently Conserving and Efficient for local conservation of mass and tracers
- Alternativly can use a posteriori correction schemes
- Need to use correctors to conserve other quantities - e.g energy

Any Questions?