Earth System Modeling
Domain decomposition

Graziano Giuliani

International Centre for Theoretical Physics
Earth System Physics Section

Advanced School on Regional Climate Modeling
over South America
February 15-19, 2015

Modelling the Atmosphere : The Equations

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \quad (1)$$

$$\frac{\partial \mathbf{u}}{\partial t} = -\nabla K - (2\Omega + \zeta) \times \mathbf{u} - \frac{1}{\rho}\nabla p + \nabla \Phi + \mathbf{F}_{ep} \quad (2)$$

$$\frac{\partial (\rho \epsilon)}{\partial t} + \nabla \cdot \left[(\rho \epsilon + p + \mathbf{F}_R) \cdot \mathbf{u} \right] = 0 \quad (3)$$

$$p = \rho R T \quad (4)$$

with:

- Mass Density ρ , Pressure p , Temperature T , Velocity $\mathbf{u}$, Gravity Potential Φ , Cinetic Energy $K = \mathbf{u}^2/2$, Relative Vorticity $\zeta = \nabla \times \mathbf{u}$
- Rotation Velocity of the Earth Ω , Ideal Gas Constant for dry air R
- Total energy per unit mass $\epsilon = c_p T + K, c_p =$ Specific heat at constant pressure, Radiation Heat Flux $\mathbf{F}_R$, Total external and parametrized forces $\mathbf{F}_{ep}$

Modeling the Atmosphere : The Problem

The set of governing equations for the atmosphere are examples of system of nonlinear partial differential equations (PDEs), to be solved on some spatial domain $\mathscr{D}$ and time interval $[0, t_f]$, given suitable boundary (BC) and initial (IC) conditions.
Therefore the typical problem to be solved in ESM is an intial/boundary value problem of the (general) form:

$$\frac{\partial \psi}{\partial t} = \mathscr{L}(\psi) \tag{5}$$

$$\psi(0) = \psi_0 \tag{6}$$

$$\psi(\mathcal{B}) = \Psi_{\mathcal{B}}(t) \tag{7}$$

Modeling the Atmosphere : The Challenge

The Problem is typically highly nonlinear. As a consequence:

- There is an interaction between different space scales.
- The system is chaotic, i.e. the solution is strongly dependent on the initial conditions. Therefore:
  - Ergodic hypothesis is made
  - Ensemble simulations are considered
- Even if you can prove that it is well posed, in general it is not possible to find a representation formula for its solution $\psi(\mathbf{x}, t)$.
- An approximation ϕ of the solution ψ has to be searched, through the introduction of suitable:
  - Space discretization techniques
    - Parametrization of unresolved sub-grid processes
  - Time integration techniques

Space Discretization

The original domain $\mathscr{D}$ is replaced by a spatially discretized domain $\mathscr{D}_h$ and the original problem is replaced by:

$$\frac{\partial \phi_i}{\partial t} = \mathscr{L}_h(\phi)_i, \quad i = 1, 2, \dots m \tag{8}$$

$$\phi_i(0) = (\phi_0)_i, \quad i = 1, 2, \dots m \tag{9}$$

$$\phi(\mathcal{B}_h) = \Phi_{\mathcal{B}_h}(t) \tag{10}$$

where:

- $\mathscr{L}_h$denotes a discrete approximation of the continuous differential operator $\mathscr{L}$
- h denotes the tipical size of the discrete spatial elements and determine the resolution of the spatial discretization.

Methods

This semi-discrete problem is a system of (generally nonlinear) Ordinary Differential Equations (ODEs) whose unknowns ϕ_i are approximations of the continuous solutions ψ.
Main approaches:

1. Finite Differences
2. Finite Volumes
3. Spectral Transform
4. Galerkin Methods / Projection methods (Finite Element, Spectral Element, Discontinuous Galerkin)

Finite Differences

- $\mathscr{D} = x_i, \quad i = 1, \ldots m$, called grid, is a regular array of discrete locations, called nodes. $h = \Delta x$ is the average spacing between nodes.
- $\phi_i(t) \approx \psi(x_i, t)$
- $\mathscr{L}_h$ is obtained by replacing derivatives present in $\mathscr{L}$ with finite difference quotients

Example in 1D:

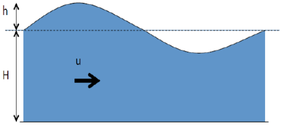
- If $\mathscr{D} = [0, L]$ and $\mathscr{L}(\psi) = \psi'$, then
  - $\mathscr{D} = \{x_i = ih, \quad i = 1, \ldots m\}$ with $h = \Delta x = L/m$
  - $\mathscr{L}_h(\phi)_i = {}^{(\phi_{i+1} - \phi_{i-1})}/_{2h}$

Shallow Water Equations

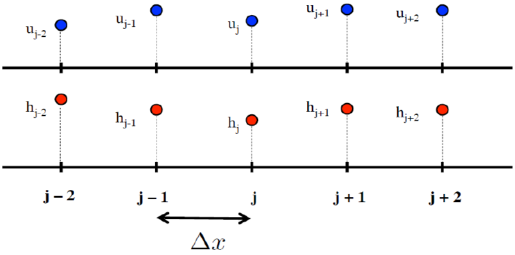
Example problem is the so called Shallow Water Equations, small amplitude wawes in a shallow basin of rest depth H. Given h the free surface wave amplitude with $h \ll H$ and g the gravity acceleration:

$$\frac{\partial h}{\partial t} + H\frac{\partial u}{\partial x} = 0 \quad (11)$$

$$\frac{\partial u}{\partial t} + g\frac{\partial h}{\partial x} = 0 \quad (12)$$



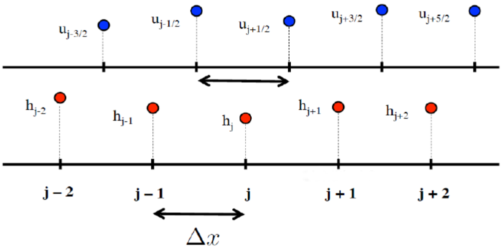
SWE discretization



$$\frac{\partial h_i}{\partial t} + H\frac{u_{i+1} - u_{i-1}}{2\Delta x} \quad (13)$$

$$\frac{\partial u_i}{\partial t} + g\frac{h_{i+1} - h_{i-1}}{2\Delta x} \quad (14)$$

Staggering in 1D

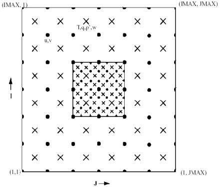


$$\frac{\partial h_i}{\partial t} + H\frac{u_{i+1/2} - u_{i-1/2}}{\Delta x} \tag{15}$$

$$\frac{\partial u_i}{\partial t} + g\frac{h_{i+1/2} - h_{i-1/2}}{\Delta x} \tag{16}$$

Staggering in 2D

Example of Arakawa B staggering with nesting:



- easy to implement
- you need to be careful to ensure shape conservation and avoid spurious wave solutions
- tipically not high order accurate
- not easy to introduce adaptivity
- heavily in Atmospheric Models

Finite Volumes

$\mathscr{D}_h = \{\mathscr{K}_i, \quad i = 1, \ldots, m\}$, called mesh, is a partition of $\mathscr{D}$ in nonoverlapping control volumes $\mathscr{K}_i$ such that $\mathscr{D} = \cup_{i=1}^{m} \mathscr{K}_i$ and $h = max_i\big(diam(\mathscr{K}_i)\big)$ is a measure of the typical size of the elements of $\mathscr{D}_h$. The approximated solution is:

$$\phi_i(t) = \frac{1}{|\mathscr{K}_i|} \int\limits_{\mathscr{K}_i} \psi(\mathbf{x}, t) d\mathbf{x} \tag{17}$$

They are inherently conservative, very robust, heavily used in industrial applications, mostly low order accurate (third order or less), but not used in atmospheric science.

Flux form and source terms

The original continuous problem is rewritten in divergence (conservation) form

$$\frac{\partial \psi}{\partial t} + \nabla \cdot \mathbf{F}(\psi) = \mathscr{S} \tag{18}$$

and integrated over each control volume $\mathscr{K}_i$. By application of Gauss theorem a set of equations is obtained for the evolution of the averages $\phi_i(t)$ in terms of their fluxes across the control volume edges:

$$\frac{d\phi_i}{dt} + \int_{\partial \mathscr{K}_i} \mathbf{F} \cdot \hat{n} d\sigma = \int_{\mathscr{K}_i} \mathscr{S} d\mathbf{x} \tag{19}$$

these fluxes are not prognostic variables but need to be recovered by appropriate interpolation procedures from the cell averaged values (subgrid scale reconstruction).

Spectral Transform Methods

They are the basis of all Global Circulation Methods.

- State Variables are not pointwise quantities, but the continuous field is written as a linear combination of modes.
- The Legendre function of first kind are a natural basis on the sphere.

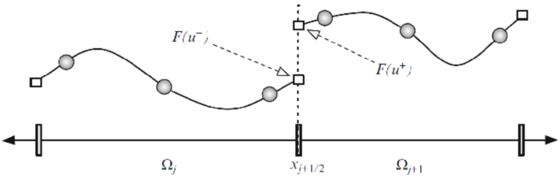
$$\psi(\mathbf{x},t) \approx \sum_{k=1}^{N} a_k(t)\phi_k(\theta,\phi) \tag{20}$$

$$\phi_k(\theta,\phi) = cos(m\phi)P_l^m(cos\theta) \tag{21}$$

$$P_n^m(\mu) = \sqrt{(2n+1)\frac{(n-m)!}{(n+m)!}}\frac{1}{2^n n!}(1-\mu^2)^{\frac{m}{2}}\frac{d^{n+m}}{d\mu^{n+m}}(\mu^2-1) \tag{22}$$

Spectral Transform Methods

Finite elements methods with the benefits of of the spectral transform methods with the locality principal of finite volume methods. Can be thought of as spectral transform in the element. Basis function in this case have compact support that can also jump at inter-element boundaries (discontinous Galerkin).



Time discretization

The space discretized time continuous solution $\phi_i(t), \quad t \in [0, t_f]$ is approximated by introducing a timestep $\Delta t = {}^{t_f}/_{n}$ and a set of discrete time levels $t_k = k\Delta t, \quad k = 0, \ldots, n$. The choice of Δt must comply with numerical CFL stability condition.
Numerical ODE's methods give the fully discrete approximated solution:

$$\phi_i^k, \quad k = 0, \ldots, n; \quad i = 1, \ldots, m \tag{23}$$

Main approches include:

1. explicit schemes
2. semi-implicit schemes
3. semi-Lagrangian shemes

Explicit Scheme

The method do not require the solution of a system at each timestep to update the solution from one discrete time level to the next: decoupled.
One of the most popular explicit time discretization is the leapfrog. It is a multistep method (three time levels scheme) derived by approximating the time derivative by a centered difference approximation:

$$\frac{\phi_i^{k+1} - \phi_i^{k-1}}{2\Delta t} = \mathscr{L}_h(\phi^k)_i \tag{24}$$

Filters can be applied to remove computational modes from the solution. Efficiency of simple explicit schemes can be improved through the "split-explicit" or "mode splitting" technique: terms responsible for the fastest waves motion are treated separately, using a smaller timestep.

Semi Implicit Scheme

They usually requires the solution of a system at each timestep to update the solution from one discrete time level to the next. The most used is the so called Crank-Nicolson scheme:

$$\frac{\phi_i^{k+1} - \phi_i^{k-1}}{2\Delta t} = \alpha \mathscr{L}_h(\phi^{k+1})_i + (1-\alpha)\mathscr{L}_h(\phi^k)_i \tag{25}$$

with $\alpha \in [0, 1]$ averaging parameter: stability is guaranteed for $\alpha \in [1/2, 1]$ and second order accuracy for $\alpha = 1/2$.

Semi Lagrangian Scheme

SL method is a discretization approach that links the spatial and time discretization for advection equations. The governing equations can be written in advective form:

$$\frac{d\psi}{dt} = \hat{\mathscr{L}}(\psi) \tag{26}$$

where ${}^{d}/_{dt} = {}^{\partial}/_{\partial t} + \mathbf{u} \cdot \nabla$ is the Lagrangian derivative. The above can be discretized as:

$$\frac{\phi_i^{k+1} - \phi_{i,*}^{k-1}}{dt} = \hat{\mathscr{L}}_h(\phi^k)_{i,*} \tag{27}$$

where $\phi_{i,*}^k$ denote an approximation of $\psi(\mathbf{x}_*, t^k)$ where $\mathbf{x}_*$ is the so-called departure point, solution at $t = t^k$ of

$$\frac{d\mathbf{X}(t, t^{k+1}, \mathbf{x}_i)}{dt} = \mathbf{u}^* \mathbf{X}(t, t^{k+1}, \mathbf{x}_i), \quad \mathbf{X}(t^{k+1}, t^{k+1}, \mathbf{x}_i) = \mathbf{x} \tag{28}$$

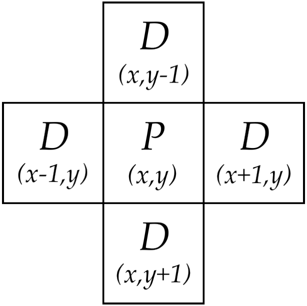
Stencil code

Atmospheric models in their dynamical core are stencil codes:

- A class of iterative kernels which update array elements according to some fixed pattern, called stencil.
- Stencil codes perform a sequence of sweeps (called timesteps) through the simulation space, a 2- or 3-dimensional regular grid whose elements are referred to as nodes, cells or elements.
- In each timestep, the stencil code updates all array elements.
- Using neighboring array elements in a fixed pattern (called the stencil), each cell's new value is computed.
- Boundary values need to be adjusted during the course of the computation as well.
- Since the stencil is the same for each element, the pattern of data accesses is repeated.

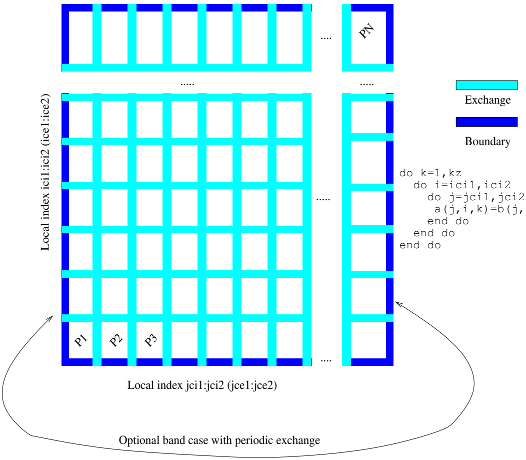
Ghost Points

The algorithm work on a patch library, which handles the synchronization of the ghost zone or halo and the boundaries.



The code loops over big arrays, and for this reason usually cannot perform efficient cache blocking or wrapping of the code for accelerators.

RegCM4 2D scheme




```
do k=1,kz
  do i=ici1,ici2
    do j=jci1,jci2
     a(j,i,k)=b(j,i,k)
    end do
  end do
end do
```

Cartesian Grid

RegCM4 works on a Cartesian 2D Grid

- Each processor has a 2D patch of the model domain
- Each processor may have exchange of ghost points
- Each processor may have boundary value area points

Cartesian Grid comunication library interface

- Different data types up to 4 dimensional
- One to all (mpi_bcast, mpi_send, mpi_recv)

```
call bcast(a)
call grid_distribute(a_glob,a_loc,j1)
call subgrid_distribute(a_glob,a_loc)
```

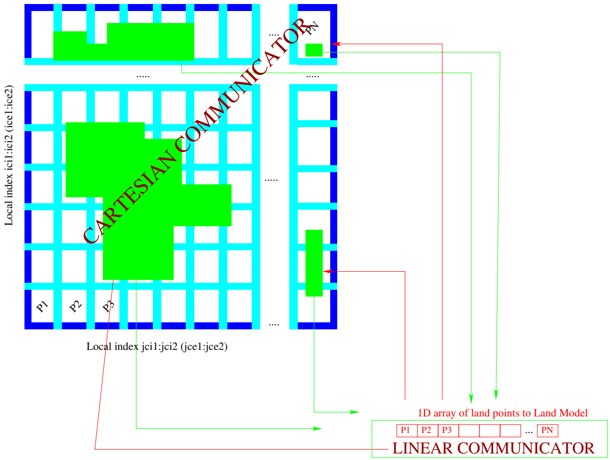
- All to one (mpi_send, mpi_recv)

```
call grid_collect(a_loc,a_glob)
call subgrid_collect(a_loc,a_glob)
```

- Ghost point exchange (mpi_irecv, mpi_send, mpi_wait, mpi_sendrecv)

```
call exchange(a)
```

RegCM4 MPI 1D surface masked scheme



Cartesian To Linear comunication interface

- Different data types up to 4 dimensional
- type(masked_comm), uses mpi_scatterv, mpi_gatherv
- Each processors gives its internal 2D grid, gets 1D land point vector. We have also the option to order up things on the global grid first.

```
call c2l_ss(masked_comm,local_matrix,vector)
call c2l_gs(masked_comm,local_matrix,vector)
call glb_c2l_ss(masked_comm,global_matrix,vector)
call glb_c2l_gs(masked_comm,global_matrix,vector)
```

- Each processor gives 1D land point vector, gets its internal 2D grid

```
call l2c_ss(masked_comm,vector,local_matrix)
call glb_l2c_ss(masked_comm,vector,global_matrix)
```

RegCM5

New Non-Hydrostatic, semi-implicit, semi-Lagrangian, p-adaptive discontinuous Galerkin method dynamical core

- Will need new parallel paradigma
  - Grid element with 3D internal quadrature nodes
  - Physical 3D Grid ($+1$ for tracers, 4D) $+$ 3D QN $=$ 7D
  - p-adaptive : variable number of QN $=$ Imbalance
  - All physical schemes do need mean vertical profile
  - New expected target grid resolution $< 10km$
  - Have an extendable Earth System Model
- Challenges
  - Efficient data types which allow load balancing
  - Efficient I/O interface to accomodate Terabytes of output
  - Use Coupling paradigm to treat multiple model components
- Usage of Cell Based Libraries