CAM-SE: Lecture II

Peter Hjort Lauritzen

Atmospheric Modeling and Predictability Section
Climate and Global Dynamics Laboratory
National Center for Atmospheric Research

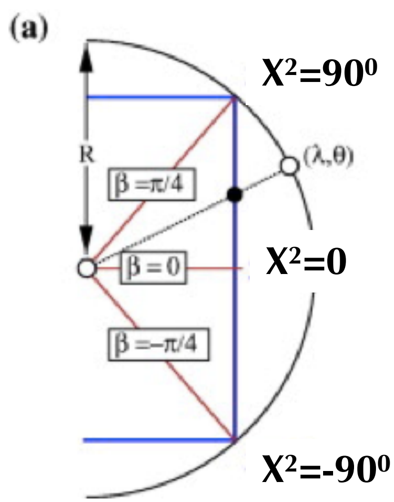
2nd WCRP Summer School on Climate Model Development: Scale aware parameterization for representing sub-grid scale processes
January 23, 2018
National Institute for Space Research, Center for Weather Forecasting and Climate Studies, Cachoeira Paulista,
São Paulo, Brazil.

NCAR UCAR | National Center for Atmospheric Research Climate & Global Dynamics climate • models • society

Outline

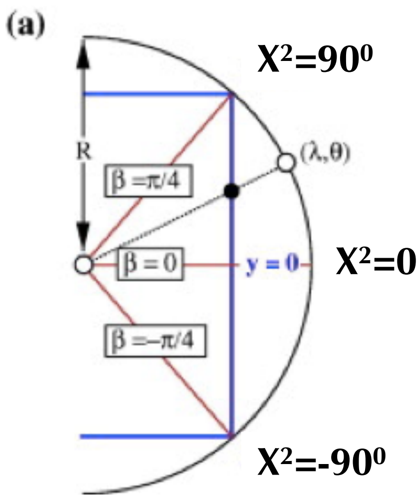
- Previous talk: derivation of equations of motion …
- Introduce computational grid and projections
- Introduction to the spectral-element method
- Properties of the spectral-element
- Time-stepping
- Coupling to physics
- Computational performance
- Variable resolution version of CAM-SE

Cubed-sphere geometry (originally introduced by Sadourny (1972))



- **A quadrilateral "box" is referred to as an element Ω_e**
- **Each 2D element $\Omega_e(x^1,x^2)$ is defined in terms of central (gnomonic) projection angles (x^1,x^2)**
- **The elements are separated by the same central angle -> equi-angular gnomonic grid**

Cubed-sphere geometry (originally introduced by Sadourny (1972))



Analytical transformation laws for the six cubed sphere faces can be found in Appendix A of Nair et al. (2005):

```
pure function cart2cubedsphere(cart3D, face_no) result(cart)
  implicit none
  type(cartesian3D_t),intent(in) :: cart3d
  integer,            intent(in) :: face_no
  type (cartesian2d_t)           :: cart

  real(kind=r8) :: x,y

  select case (face_no)
  case (1)
     x =  cart3D%y/cart3D%x
     y =  cart3D%z/cart3D%x
  case (2)
     x = -cart3D%x/cart3D%y
     y =  cart3D%z/cart3D%y
  case (3)
     x =  cart3D%y/cart3D%x
     y = -cart3D%z/cart3D%x
  case (4)
     x = -cart3D%x/cart3D%y
     y = -cart3D%z/cart3D%y
  case (5)
     x  = -cart3D%y/cart3D%z
     y  = -cart3D%x/cart3D%z
  case (6)
     x  =  cart3D%y/cart3D%z
     y  = -cart3D%x/cart3D%z
  end select
  cart%x = ATAN(x)
  cart%y = ATAN(y)
end function cart2cubedsphere
```

```
pure function spherical_to_cart(sphere) result (cart)
  implicit none
  type(spherical_polar_t), intent(in) :: sphere
  type(cartesian3D_t)                 :: cart

  cart%x=sphere%r*COS(sphere%lat)*COS(sphere%lon)
  cart%y=sphere%r*COS(sphere%lat)*SIN(sphere%lon)
  cart%z=sphere%r*SIN(sphere%lat)

end function spherical_to_cart
```

Mapping a vector from cube to sphere

The mapping from cube to sphere results in a non-orthogonal curvilinear coordinate system on $\mathcal{S}$, with the metric tensor G_{ij} and analytic Jacobian $\sqrt{G} = |G_{ij}|^{1/2}$, $i, j \in \{1, 2\}$. A physical vector quantity such as the wind vector $\mathbf{v} = (u, v)$, defined on $\mathcal{S}$ in orthogonal lat-lon coordinates, can be uniquely expressed in tensor form using conventional notations as the covariant (u_1, u_2) and contravariant (u^1, u^2) vectors using the 2×2 transformation matrix $\mathbf{D}$ associated with the gnomonic mapping such that $\mathbf{D}^T\mathbf{D} = G_{ij}$ (see *Nair et al.* [2005] for the details):

$$\begin{bmatrix} u \\ v \end{bmatrix} = \mathbf{D} \begin{bmatrix} u^1 \\ u^2 \end{bmatrix} = \mathbf{D}^{-T} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}.$$

Outline

- Previous talk: derivation of equations of motion …
- Introduce computational grid and projections
- **Introduction to the spectral-element method**
- Properties of the spectral-element
- Time-stepping
- Coupling to physics
- Computational performance
- Variable resolution version of CAM-SE

The spectral element method: 2D conservation law

The governing equations defined in familiar vector form can also be expressed in general tensor form. In order to describe the SE discretization process in simple terms, we consider the the following conservation law on $\mathcal{S}$ for an arbitrary scalar ϕ:

$$\frac{\partial \phi}{\partial t} + \nabla \cdot \mathbf{F}(\phi) = S(\phi), \tag{76}$$

where

$$\nabla \cdot \mathbf{F}(\phi) = \frac{1}{\sqrt{G}} \left[\frac{\partial \sqrt{G} F^1}{\partial x^1} + \frac{\partial \sqrt{G} F^2}{\partial x^2} \right]. \tag{77}$$

In the special case of the flux-form transport equation the contravariant fluxes $(F^1, F^2) = (u^1\phi, u^2\phi)$, and $S(\phi)$ is an arbitrary source term.

The spectral element method: test function

The SE solution process involves casting the partial differential equation in Galerkin form, i.e., by multiplying (76) with a test (weight) function ψ and integrating over the domain $\mathcal{S}$,

$$\int_{\mathcal{S}} \psi \left[\frac{\partial \phi}{\partial t} + \nabla \cdot \mathbf{F}(\phi) - S(\phi) \right] d\mathcal{S} = 0. \tag{78}$$

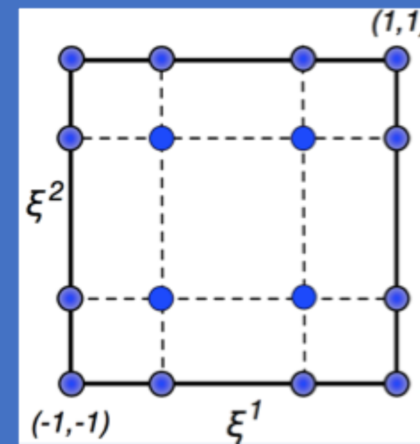
A computational form of (78) is obtained by applying Green's theorem, resulting in the weak Galerkin form, as follows:

$$\int_{\mathcal{S}} \psi \, \frac{\partial \phi}{\partial t} \, d\mathcal{S} = \int_{\mathcal{S}} \nabla \psi \cdot \mathbf{F}(\phi) \, d\mathcal{S} + \int_{\mathcal{S}} \psi \, S(\phi) \, d\mathcal{S}, \tag{79}$$

where the approximation to the solution ϕ and the test function belong to a polynomial space V^N.

For the efficient evaluation of the integrals, the SE method employs Gauss-Lobatto-Legendre (GLL) quadrature rule for integrals and collocation differentiation for derivative operators.

All the corresponding numerical operations are performed on a square $[-1, 1]^2$ known as the standard (or reference) element.



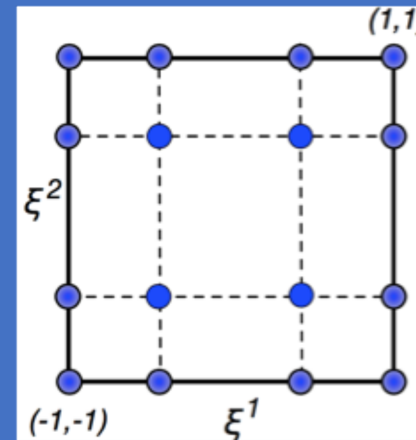
A computational form of (78) is obtained by applying G ing in the weak Galerkin form, as follows:

$$\int_S \psi \, \frac{\partial \phi}{\partial t} \, dS = \int_S \nabla \psi \cdot \mathbf{F}(\phi) \, dS + \int_S \psi \, S(\phi) \, dS, \tag{79}$$

where the approximation to the solution ϕ and the test function belong to a polynomial space V^N.

For the efficient evaluation of the integrals, the SE method employs Gauss-Lobatto-Legendre (GLL) quadrature rule for integrals and collocation differentiation for derivative operators.

All the corresponding numerical operations are performed on a square $[-1, 1]^2$ known as the standard (or reference) element.



Then an arbitrary surface integral on Ω_e can be expressed in terms of local coordinates $\xi^1, \xi^2 \in [-1, 1]$ and the Jacobian J_e

$$\int_{\Omega_e} \psi(x^1, x^2)\, d\Omega_e = \int_{-1}^{1} \int_{-1}^{1} J_e(\xi^1, \xi^2)\, \psi(\xi^1, \xi^2)\, d\xi^1\, d\xi^2 \approx \sum_{k=0}^{N} \sum_{l=0}^{N} w_k w_l\, J_e(\xi_k^1, \xi_l^2)\, \psi(\xi_k^1, \xi_l^2), \tag{80}$$

where w_k, w_l are the Gauss quadrature weights.

In the case of GLL quadrature rule, the nodal points ξ_k, $k = 0, 1, \ldots, N$, are the roots of the polynomial $(1 - \xi^2)P'_N(\xi) = 0$, $\xi \in [-1, 1]$; and the corresponding GLL quadrature weights are given by

$$w_k = \frac{2}{N(N+1)\,[P_N(\xi_k)]^2},$$

where $P_N(\xi)$ is the Legendre polynomial of degree N.

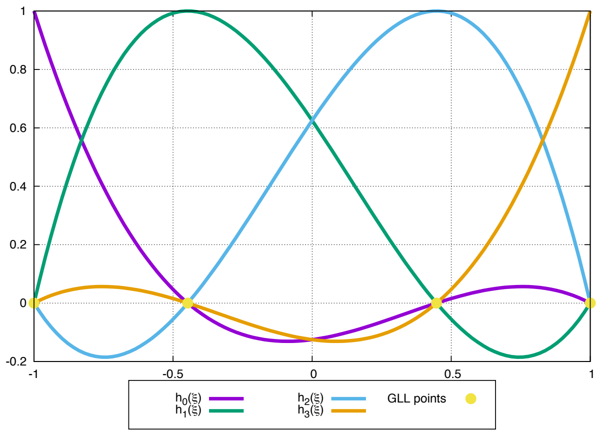
Note that there are N+1 GLL point

Then an arbitrary surface integral on [illegible] expressed in terms of local coordinates $\xi^1, \xi^2 \in [-1, 1]$ and the Jacobian J_e

$$\int_{\Omega_e} \psi(x^1, x^2)\, d\Omega_e = \int_{-1}^{1}\int_{-1}^{1} J_e(\xi^1, \xi^2)\, \psi(\xi^1, \xi^2)\, d\xi^1\, d\xi^2 \approx \sum_{k=0}^{N}\sum_{l=0}^{N} w_k w_l\, J_e(\xi_k^1, \xi_l^2)\, \psi(\xi_k^1, \xi_l^2), \tag{80}$$

where w_k, w_l are the Gauss quadrature weights.

For the SE discretization it is customary to use Lagrange polynomials $h_k(\xi)$, with roots at the GLL quadrature points ξ_k, as basis functions. This setup provides discrete orthogonality for the basis function $h_k(\xi)$, which is formally defined as:



$$h_k(\xi) = \frac{(\xi^2 - 1)\, P'_N(\xi)}{N(N+1)\, P_N(\xi_k)\, (\xi - \xi_k)}.$$

$$\int_{\Omega_e} \psi(x^1, x^2)\, d\Omega_e = \int_{-1}^{1}\int_{-1}^{1} J_e(\xi^1, \xi^2)\, \psi(\xi^1, \xi^2)\, d\xi^1\, d\xi^2 \approx \sum_{k=0}^{N}\sum_{l=0}^{N} w_k w_l\, J_e(\xi^1_k, \xi^2_l)\, \psi(\xi^1_k, \xi^2_l), \tag{80}$$

where w_k, w_l are the Gauss quadrature weights.

The spectral element method: semi-discrete form

$$\int_S \psi \frac{\partial \phi}{\partial t} dS = \int_S \nabla\psi \cdot \mathbf{F}(\phi)\, dS + \int_S \psi\, S(\phi)\, dS, \tag{79}$$

A semi-discrete form of (79) on an element Ω_e can by obtained by approximating the solution as a tensor product of 1D Lagrange basis $\{h_k(\xi)\}_{k=0}^{N}$ such that

$$\phi|_{\Omega_e} \approx \phi^e(\xi^1, \xi^2, t) = \sum_{k=0}^{N} \sum_{l=0}^{N} \phi^e_{kl}(t)\, h_k(\xi^1)\, h_l(\xi^2),$$

where $\phi^e_{kl}(t) = \phi^e(\xi^1_k, \xi^2_l, t)$ are the nodal grid-point values of the solution, and defining the test function as $\psi(\xi^1, \xi^2) = h_k(\xi^1)\, h_l(\xi^2)$.

The spectral element method: semi-discrete form

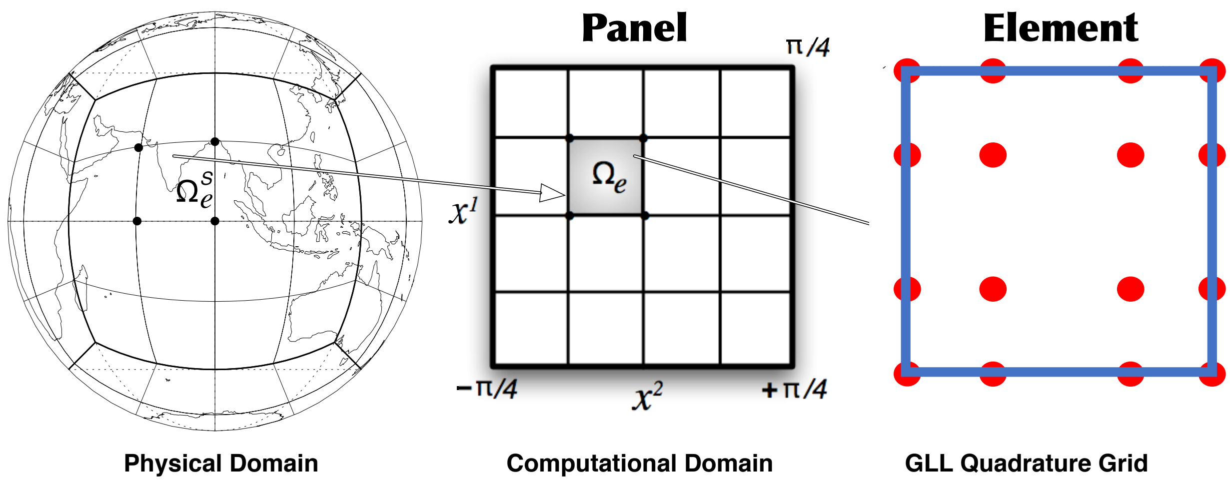
$$\int_S \psi \frac{\partial \phi}{\partial t} dS = \int_S \nabla\psi \cdot \mathbf{F}(\phi)\, dS + \int_S \psi\, S(\phi)\, dS, \qquad (79)$$

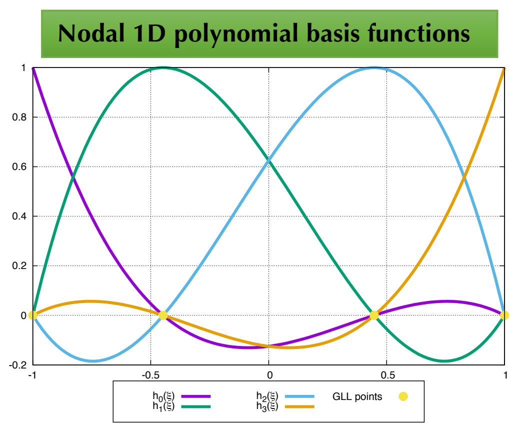
By using (80) and the discrete orthogonality property of $h_k(\xi)$, we get a completely decoupled system of ODEs on Ω_e, for each grid-point (k, l)

$$\begin{aligned}
\widetilde{M}^e_{kl} \frac{d}{dt}\phi^e_{kl}(t) &= A^e_{kl} + S^e_{kl} \\
\widetilde{M}^e_{kl} &= \int_{-1}^{1}\int_{-1}^{1} J_e\, h_k(\xi^1)\, h_l(\xi^2)\, d\xi^1 d\xi^2 = J_e(k,l)\, w_k\, w_l \\
A^e_{kl} &= \sum_{i=0}^{N} J_e^{(1)}(i,l)\, F^1_{il}\, D^{(1)}_{ik}\, w_i w_l + \sum_{i=0}^{N} J_e^{(2)}(k,i)\, F^2_{ki}\, D^{(2)}_{li}\, w_k w_i \\
S^e_{kl} &= J_e(k,l)\, w_k w_l\, S(U_{kl})
\end{aligned}$$

where $J_e^{(i)} = J_e\, \partial\xi^i/\partial x^i$ is the metric term and $D^{(i)}_{lk}$ is the derivative matrix $h'_k(\xi^i_l)$, along the x^i-direction and $i \in \{1, 2\}$.

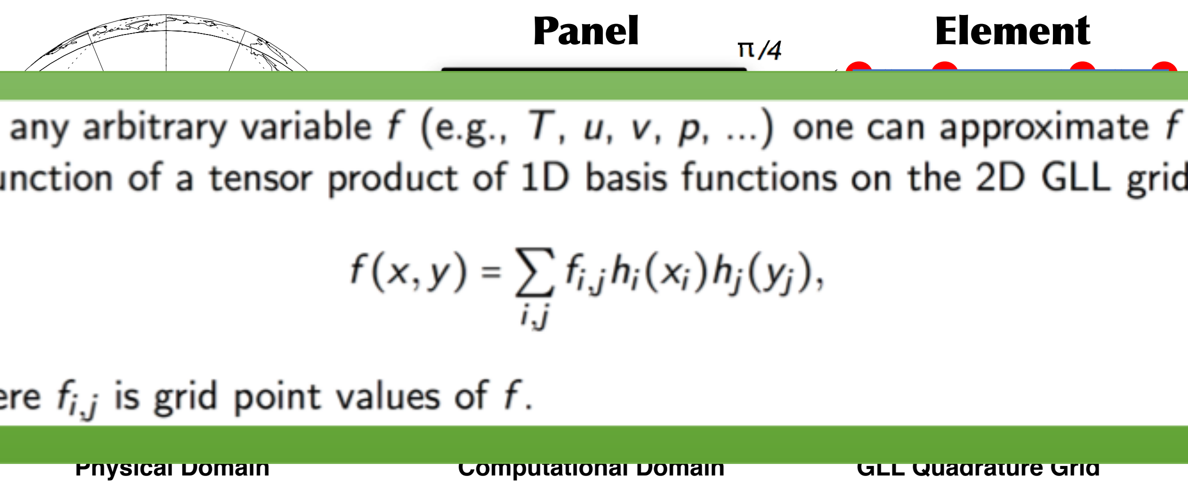
Story so far: solving equations of motion on each element





GLL=Gauss-Lobatto-Legendre

Story so far: solving equations of motion on each element



For any arbitrary variable f (e.g., T, u, v, p, ...) one can approximate f as a function of a tensor product of 1D basis functions on the 2D GLL grid:

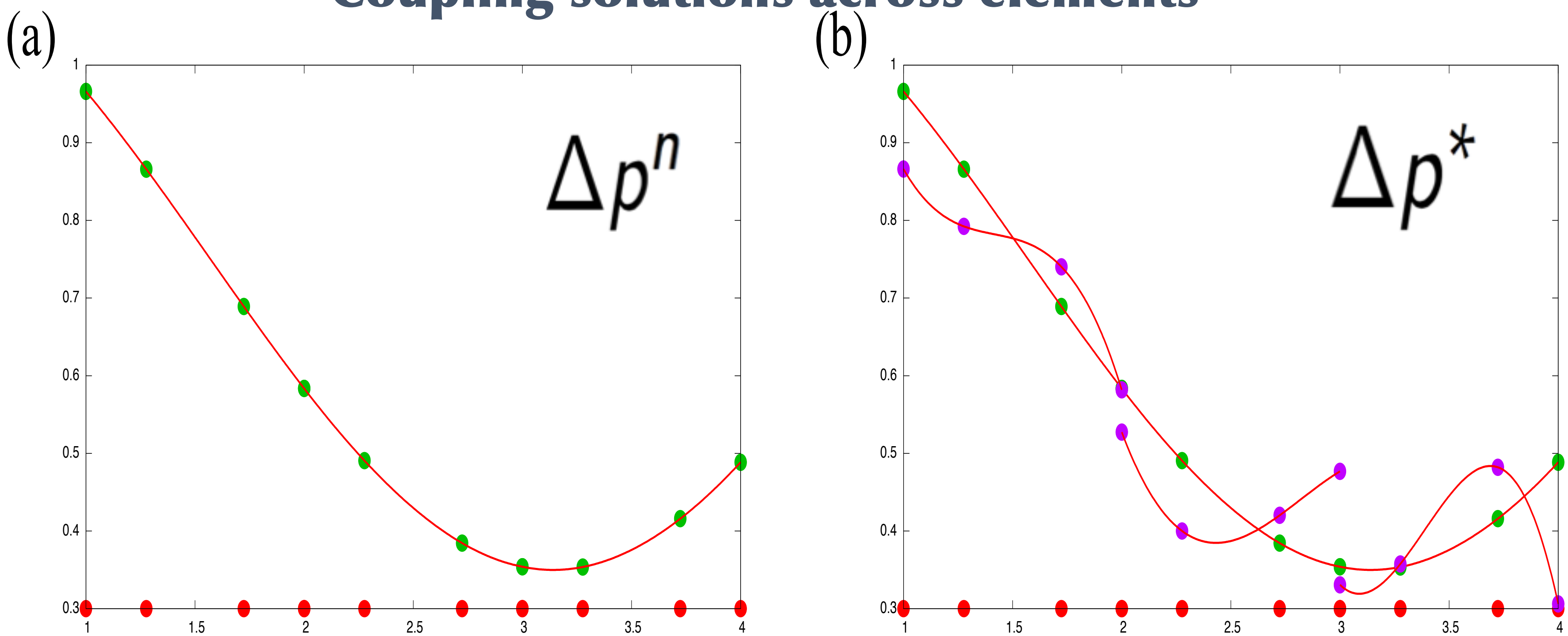
$$f(x,y) = \sum_{i,j} f_{i,j} h_i(x_i) h_j(y_j),$$

where $f_{i,j}$ is grid point values of f.

Nodal 1D polynomial basis functions

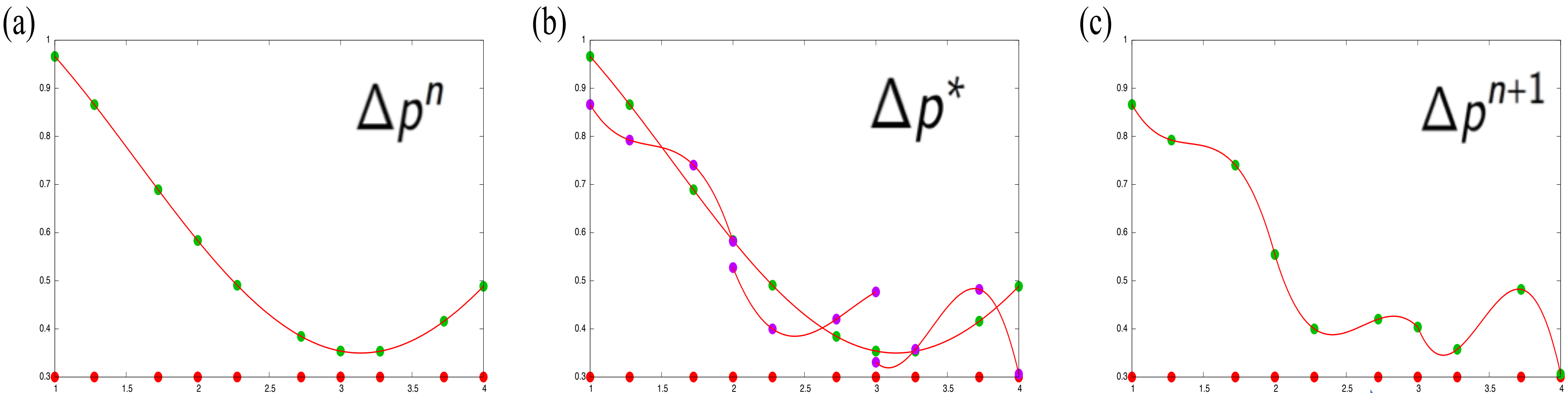
GLL=Gauss-Lobatto-Legendre

Coupling solutions across elements



Advance solution in each element one Runge Kutta step

Coupling solutions across elements



Global C^0 polynomial

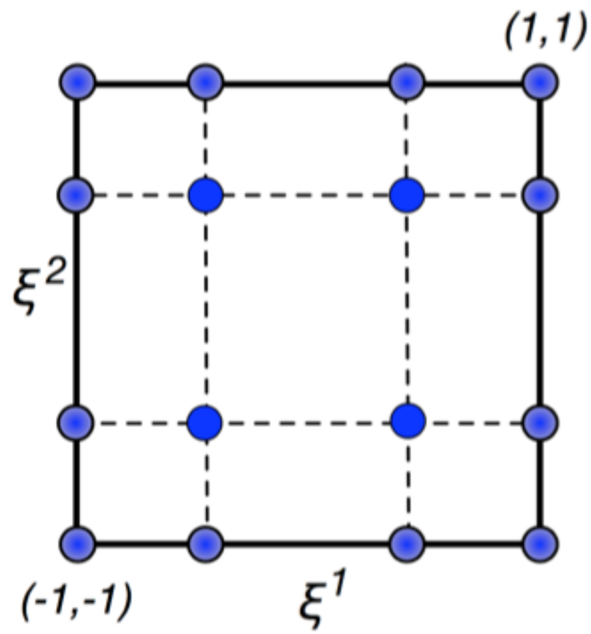
- Projection step

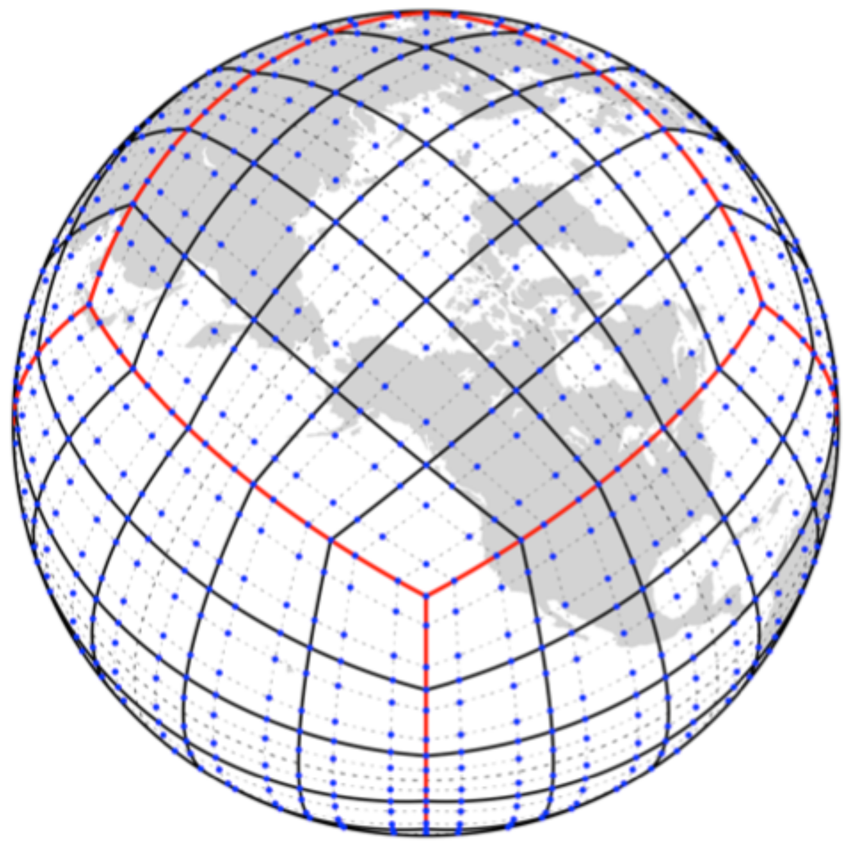
$$\Delta p^{n+1} = DSS\,(\Delta p^{*})$$

  where *DSS* refers to *Direct Stiffness Summation* (also referred to as assembly or inverse mass matrix step).

- Choice of GLL quadrature based inner product and nodal basis functions gives a diagonal mass matrix (Maday and Patera, 1987).

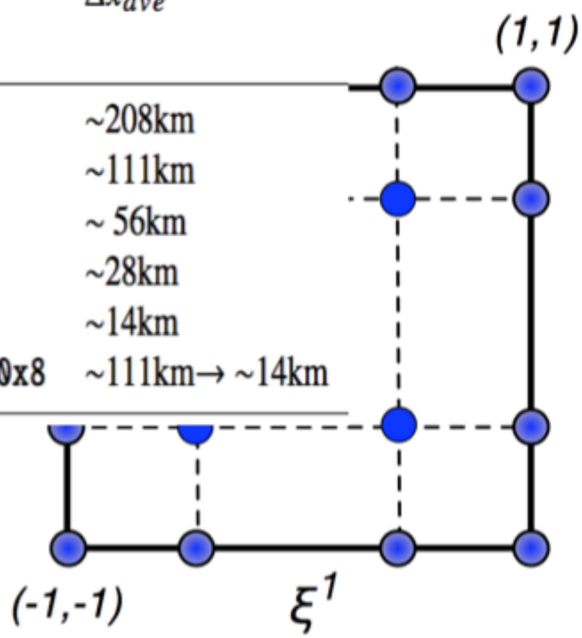
Specification of resolution in CAM:
number of GLL points is always 4x4 in each element
Number of elements determines resolution

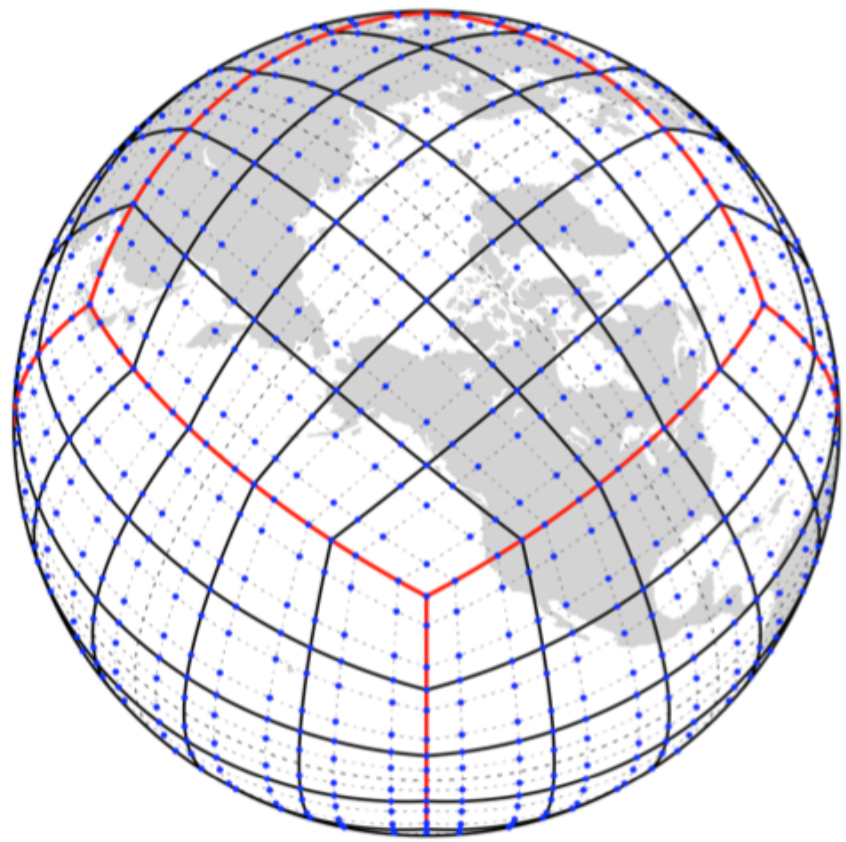




Specification of resolution in CAM:
number of GLL points is always 4x4 in each element
Number of elements determines resolution

Horizontal res.	Δx_{ave}
ne16np4	~208km
ne30np4	~111km
ne60np4	~ 56km
ne120np4	~28km
ne240np4	~14km
ne0np4CONUS30x8	~111km→ ~14km





Properties of the SE method

A numerical method is mimetic (or compatible) if key integral properties of divergence, gradient and curl-operators are mimicked in discretized space. The CAM-SE discretization satisfies the divergence/gradient adjoint relation

In the floating Lagrangian layer CAM-SE conserves total energy to time truncation errors and conserves mass

$$\int \phi \nabla \cdot \mathbf{v} \, d\mathcal{S} + \int \mathbf{v} \cdot \nabla \phi \, d\mathcal{S} = 0$$

in discretized space [*Taylor and Fournier*, 2010]. This property can be used to show the inherent conservation properties of CAM-SE in terms of mass and energy in the horizontal discretization. This is discussed in detail in *Taylor* [2011] and hence not repeated here.

Outline

- Previous talk: derivation of equations of motion …
- Introduce computational grid and projections
- Introduction to the spectral-element method
- Properties of the spectral-element
- **Time-stepping**
- Coupling to physics
- Computational performance
- Variable resolution version of CAM-SE

Time-stepping (u,v,T,dM$^{(d)}$)

The Kinnmark-Gray 5-stage 3rd-order Runge-Kutta scheme

The thermodynamic equation, momentum equations and dry air-mass continuity equation in CAM-SE are evolved using the 5-stage 3rd-order Runge-Kutta scheme described in *Guerra and Ullrich* [2016, see their equation (56)]. The stability of this class of time-stepping schemes is discussed in *Dubos et al.* [2015, see their section 3.4]. For a given initial state vector at timestep n, $\vec{\Lambda}^{(0)} = \vec{\Lambda}^{n}$, the updated state vector $\vec{\Lambda}^{(5)} = \vec{\Lambda}^{n+1}$ is computed as follows:

$$
\begin{aligned}
\vec{\Lambda}^{(1)} &= \vec{\Lambda}^{(0)} + \tfrac{\Delta t}{5}\vec{\Upsilon}(\vec{\Lambda}^{(0)}),\\
\vec{\Lambda}^{(2)} &= \vec{\Lambda}^{(0)} + \tfrac{\Delta t}{5}\vec{\Upsilon}(\vec{\Lambda}^{(1)}),\\
\vec{\Lambda}^{(3)} &= \vec{\Lambda}^{(0)} + \tfrac{\Delta t}{3}\vec{\Upsilon}(\vec{\Lambda}^{(2)}),\\
\vec{\Lambda}^{(4)} &= \vec{\Lambda}^{(0)} + \tfrac{2\Delta t}{3}\vec{\Upsilon}(\vec{\Lambda}^{(3)}),\\
\vec{\Lambda}^{(5)} &= -\tfrac{1}{4}\vec{\Lambda}^{(0)} + \tfrac{5}{4}\vec{\Lambda}^{(1)} + \tfrac{3\Delta t}{4}\vec{\Upsilon}(\vec{\Lambda}^{(4)}),
\end{aligned}
$$

where $\vec{\Upsilon}(\vec{\Lambda})$ denotes the discrete right-hand-side terms of the equations of motion. The resulting method is linearly and non-linearly third-order accurate.

Time-stepping (u,v,T,dM$^{(d)}$)

The scheme possesses a stability region which is provably optimal in terms of its extent along the imaginary axis among all 5-stage 3rd-order Runge-Kutta schemes, $[-i\sqrt{15}, i\sqrt{15}]$. Since the largest eigenvalue of the 1D 4th-order spectral element spatial discretization ($N_p = 4$) is $i\sqrt{10/3}$, the resulting scheme satisfies a Courant-Friedrichs-Lewy condition given by

$$\frac{c\Delta t}{\Delta x} \leq \frac{3}{\sqrt{2}}, \qquad \text{(1D condition)},$$

where Δx denotes the average distance between degrees of freedom (equal to $3 \times \Delta x_e$, the width of a spectral element), c is the gravity wave speed, and Δt is the timestep size. As dimension splitting is not employed in CAM-SE, in 2D and on a uniformly spaced grid this condition is restricted by a further factor of $1/\sqrt{2}$ to

$$\frac{c\Delta t}{\Delta x} \leq \frac{3}{2}, \qquad \text{(2D condition)}.$$

Time-stepping (tracers)

- Continuity equation for air is coupled with momentum and thermodynamic equations:
  - thermodynamic variables and other prognostic variables feed back on the velocity field
  - which, in turn, feeds back on the solution to the continuity equation.
  - Hence the continuity equation for air can not be solved in isolation and one must obey the maximum allowable time-step restrictions imposed by the fastest waves in the system.
- The tracer transport equation can be solved in isolation given prescribed winds and air densities, and is therefore not susceptible to the time-step restrictions imposed by the fastest waves in the system.
- For efficiency: Use longer time-step for continuity equation for tracers than for air.

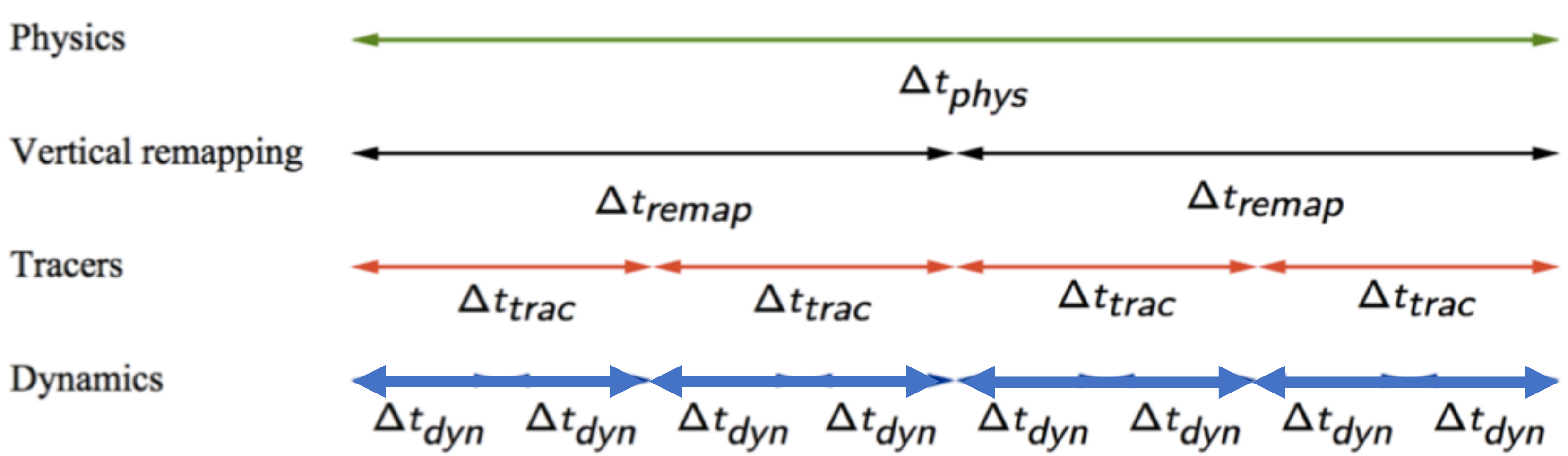
Or different time-stepping or different numerical method

Time-stepping (tracers)

- Continuity equation for air is coupled with momentum and thermodynamic equations:
  - thermodynamic variables and other prognostic variables feed back on the velocity field
  - which, in turn, feeds back on the solution to the continuity equation.
  - Hence the continuity equation for air can not be solved in isolation and one must obey the maximum allowable time-step restrictions imposed by the fastest waves in the system.
- The tracer transport equation can be solved in isolation given prescribed winds and air densities, and is therefore not susceptible to the time-step restrictions imposed by the fastest waves in the system.
- For efficiency: Use longer time-step for continuity equation for tracers than for air.

The SE tracer advection algorithm uses a three-stage RK strong-stability-preserving (SSP) time-stepping method, ensuring the time step will preserve any shape-preserving properties preserved by the underlying spatial discretization [*Spiteri and Ruuth*, 2002]. The shape-preserving filter used is described in *Guba et al.* [2014b]. The shape-preserving SE tracer advection algorithm is formally second-order accurate.

Time-steps in CAM-SE



Time-steps in CAM-SE

$$
\begin{aligned}
\Delta t_{remap} &= \frac{\Delta t_{phys}}{\texttt{se\_nsplit}}, \\
\Delta t_{tracer} &= \frac{\Delta t_{phys}}{\texttt{se\_nsplit} * \texttt{se\_qsplit}}, \\
\Delta t_{dyn} &= \frac{\Delta t_{phys}}{\texttt{se\_nsplit} * \texttt{se\_rsplit} * \texttt{se\_qsplit}}, \\
\Delta t_{hyper} &= \frac{\Delta t_{phys}}{\texttt{se\_nsplit} * \texttt{se\_rsplit} * \texttt{se\_hypervis\_subcycle} * \texttt{se\_qsplit}}
\end{aligned}
$$

Outline

- Previous talk: derivation of equations of motion …
- Introduce computational grid and projections
- Introduction to the spectral-element method
- Properties of the spectral-element
- Time-stepping
- **Coupling to physics**
- Computational performance
- Variable resolution version of CAM-SE

Physics dynamics coupling methods in CAM-SE: se_ftype=1

Advance dynamics core (30 minutes)

Compute physics tendencies based on dynamics updated state

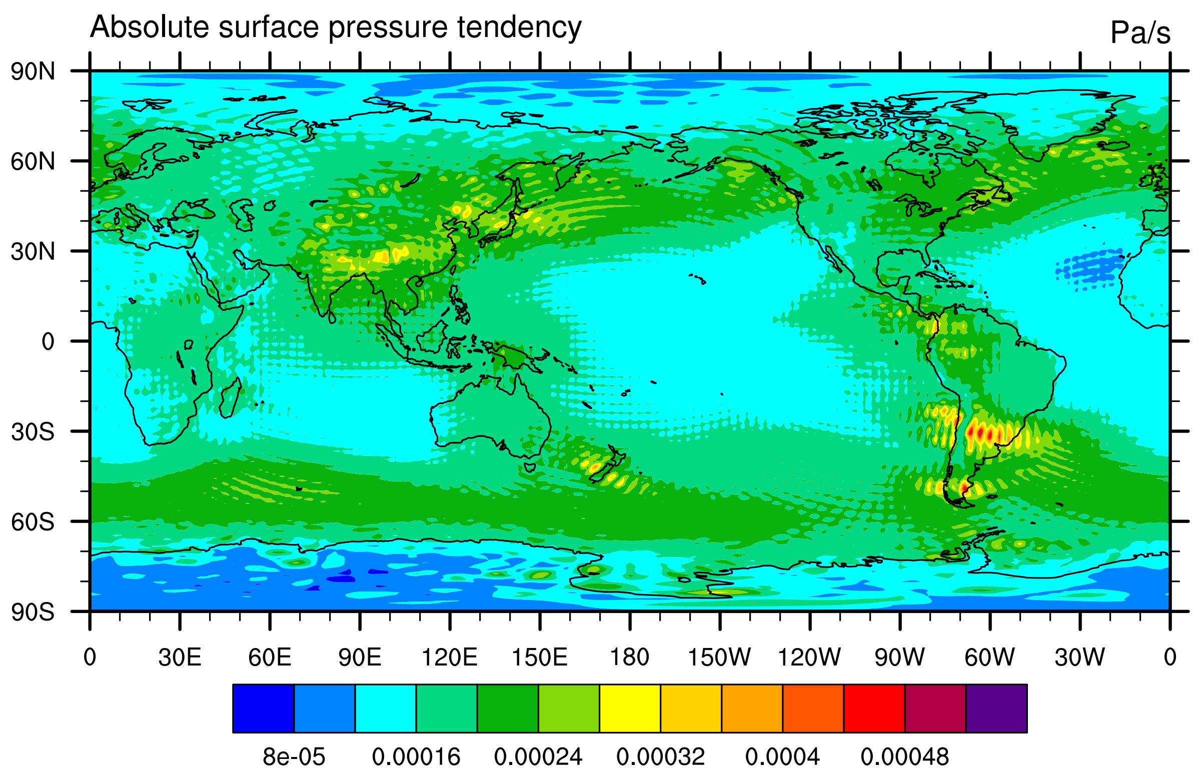
Update dynamics state with physics tendencies

For long physics time-steps and less diffusive dynamical cores this can create spurious noise!

Noise can be detected by computing

$$\frac{d}{dt}|p_s|$$

10 year average of $\frac{d}{dt}|p_s|$ from AMIP run



NCAR UCAR | National Center for Atmospheric Research Climate & Global Dynamics climate • models • society

Physics dynamics coupling methods in CAM-SE: se_ftype=0

Advance dynamics core (30 minutes): add physics tendency "chunks" during the dynamics time-stepping - every 15 minutes in this example (I refer to it as "dribbling")

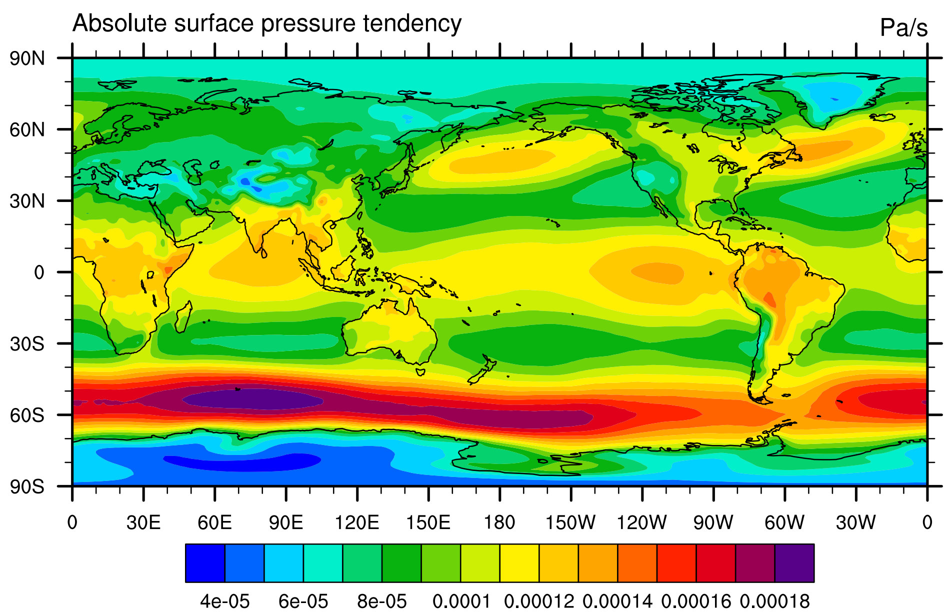
Compute physics tendencies based on dynamics updated state

Split physics tendencies into a number of "chunks"

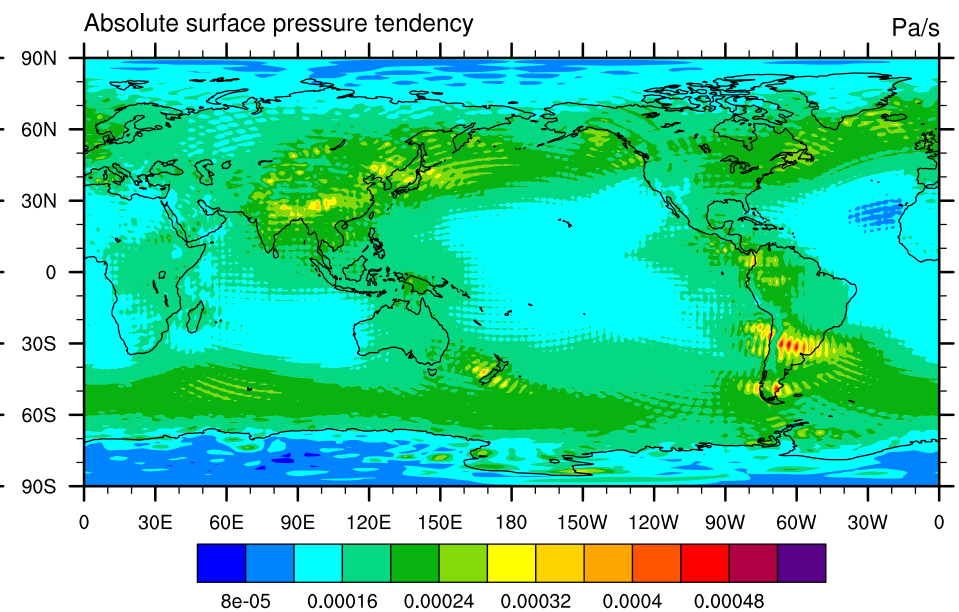
Physics dynamics coupling methods in CAM-SE: se_ftype=1

10 year average of $\frac{d}{dt}|p_s|$ from AMIP run

“Dribbling” physics tendencies



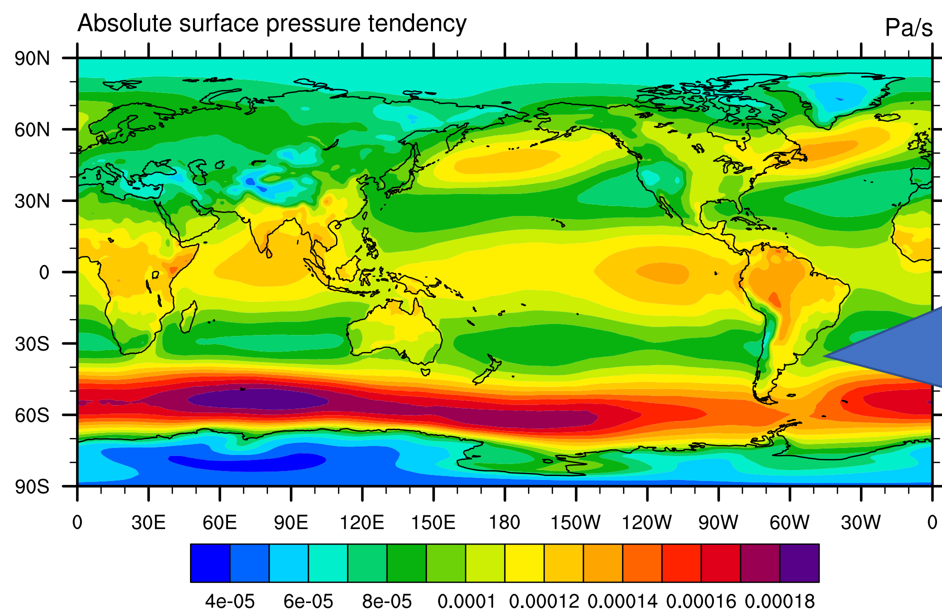
State updated every 30 minutes



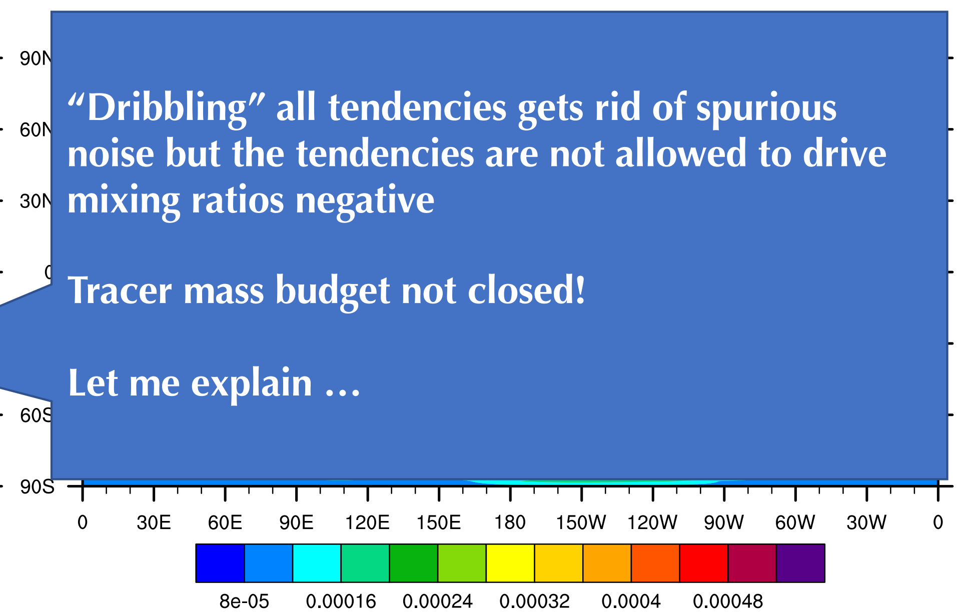
Physics dynamics coupling methods in CAM-SE: se_ftype=1

10 year average of $\frac{d}{dt}|p_s|$ from AMIP run

"Dribbling" physics tendencies



State updated every 30 minutes

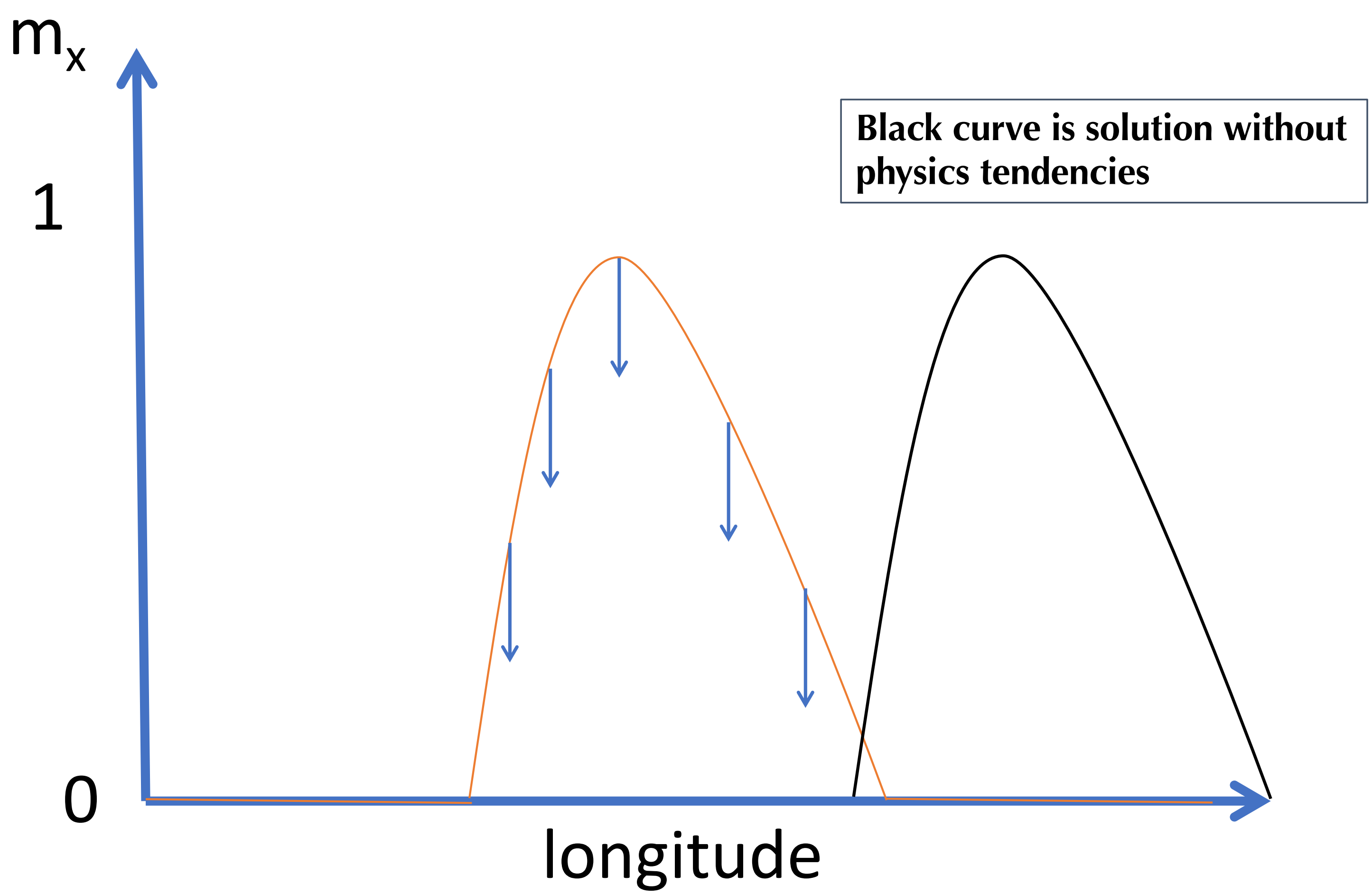


"Dribbling" all tendencies gets rid of spurious noise but the tendencies are not allowed to drive mixing ratios negative

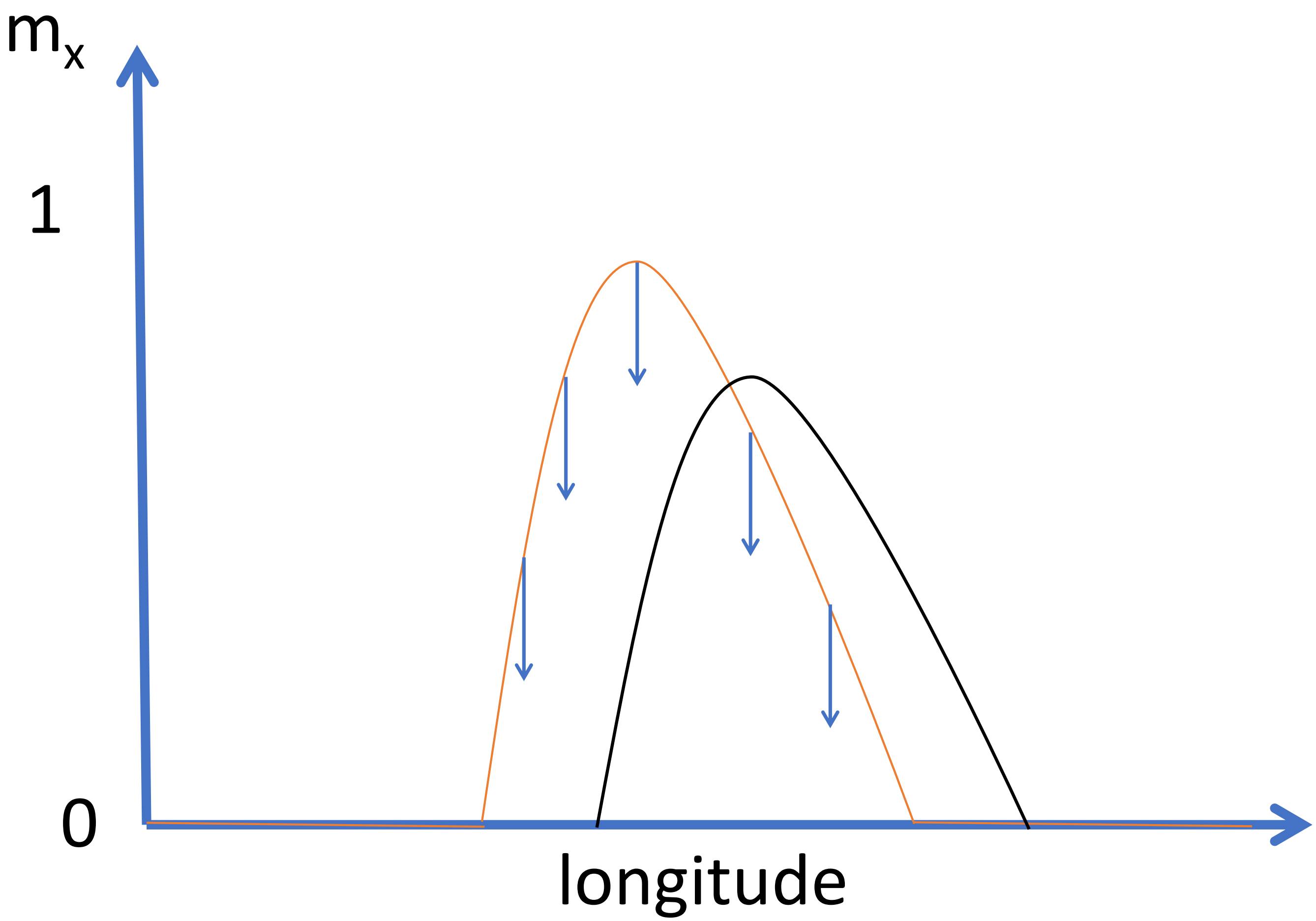
Tracer mass budget not closed!

Let me explain …

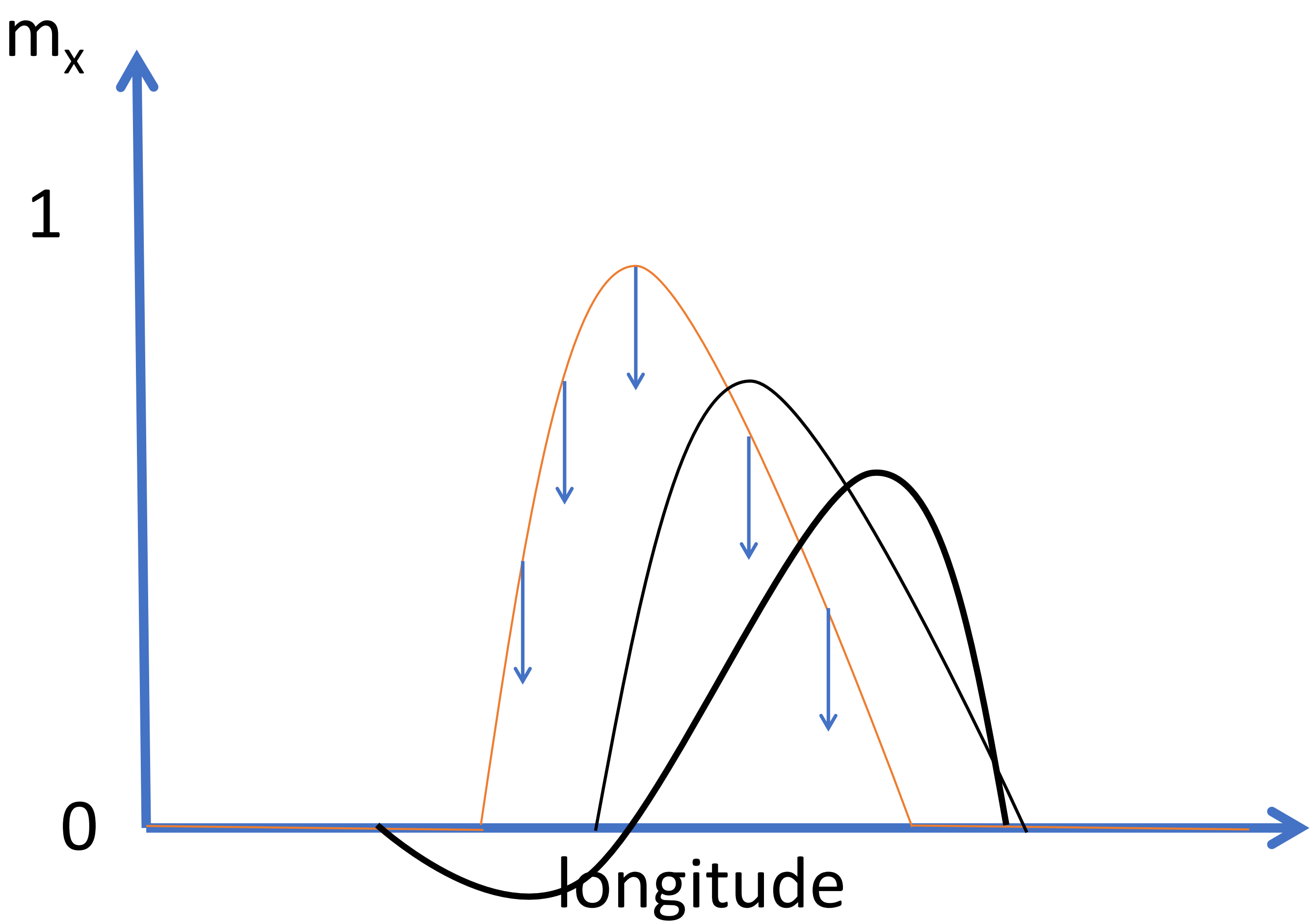
Physics-dynamics coupling: “dribbling” tendencies



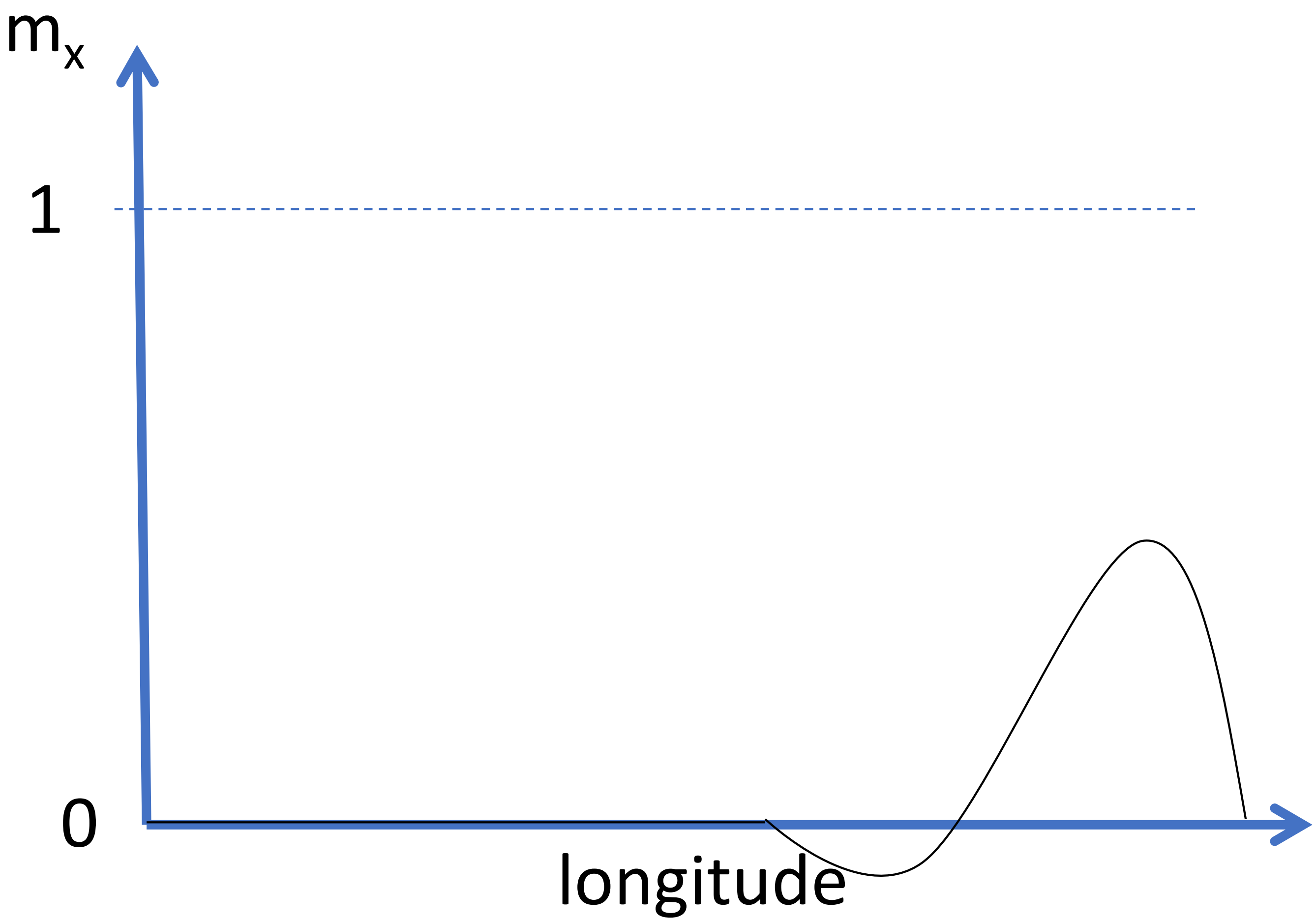
Physics-dynamics coupling: "dribbling" tendencies



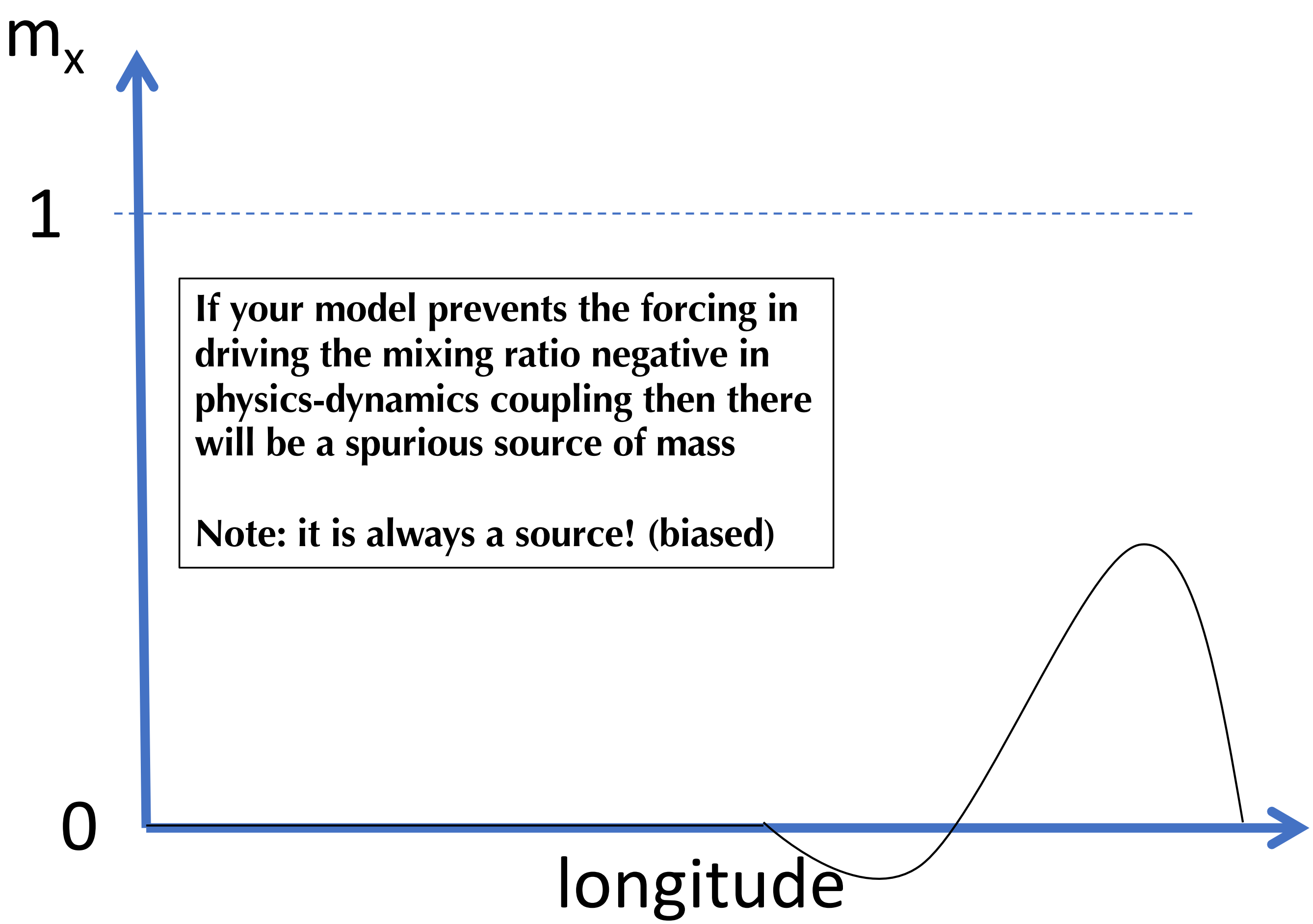
Physics-dynamics coupling: "dribbling" tendencies



Physics-dynamics coupling: "dribbling" tendencies

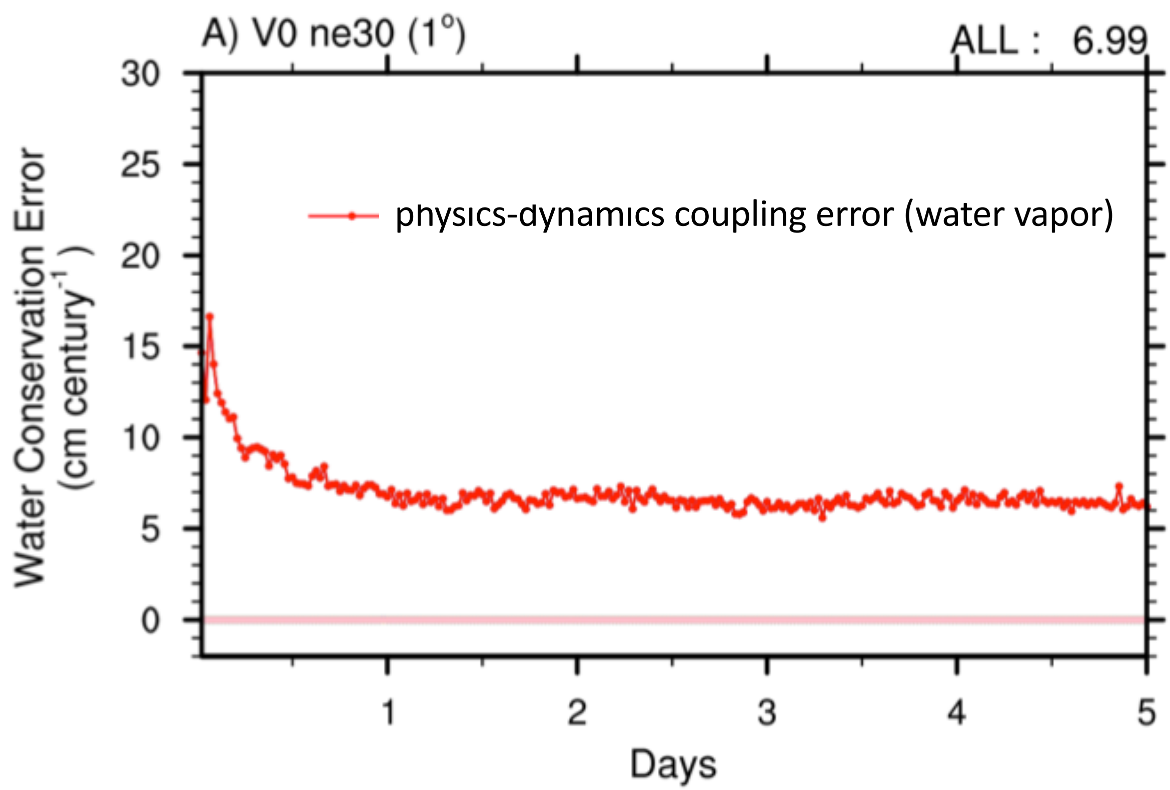


Physics-dynamics coupling: “dribbling” tendencies



Physics-dynamics coupling: "dribbling" tendencies

Example from fully coupled climate model: water conservation errors

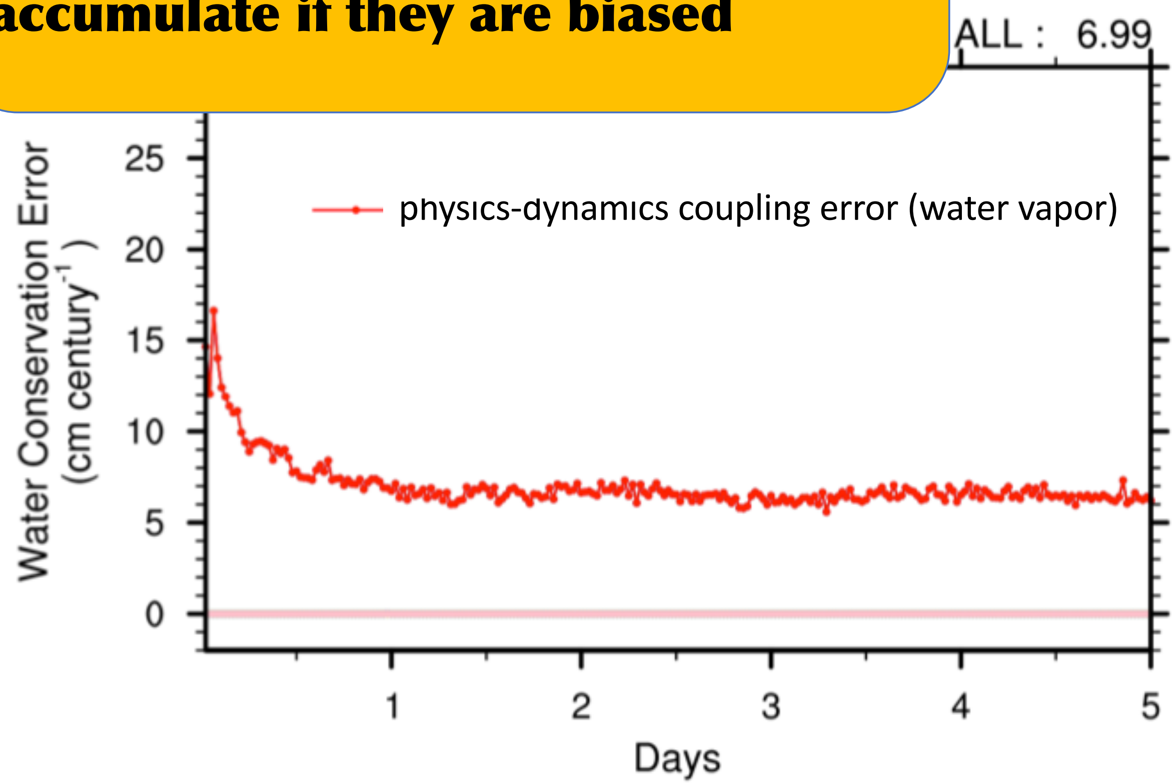


Geosci. Model Dev. Discuss., https://doi.org/10.5194/gmd-2017-293
Manuscript under review for journal Geosci. Model Dev.
Discussion started: 15 December 2017

…ing" tendencies

…errors

Note: The time-step by time-step error is miniscule but in long simulations tiny errors can accumulate if they are biased



Geosci. Model Dev. Discuss., https://doi.org/10.5194/gmd-2017-293
Manuscript under review for journal Geosci. Model Dev.
Discussion started: 15 December 2017

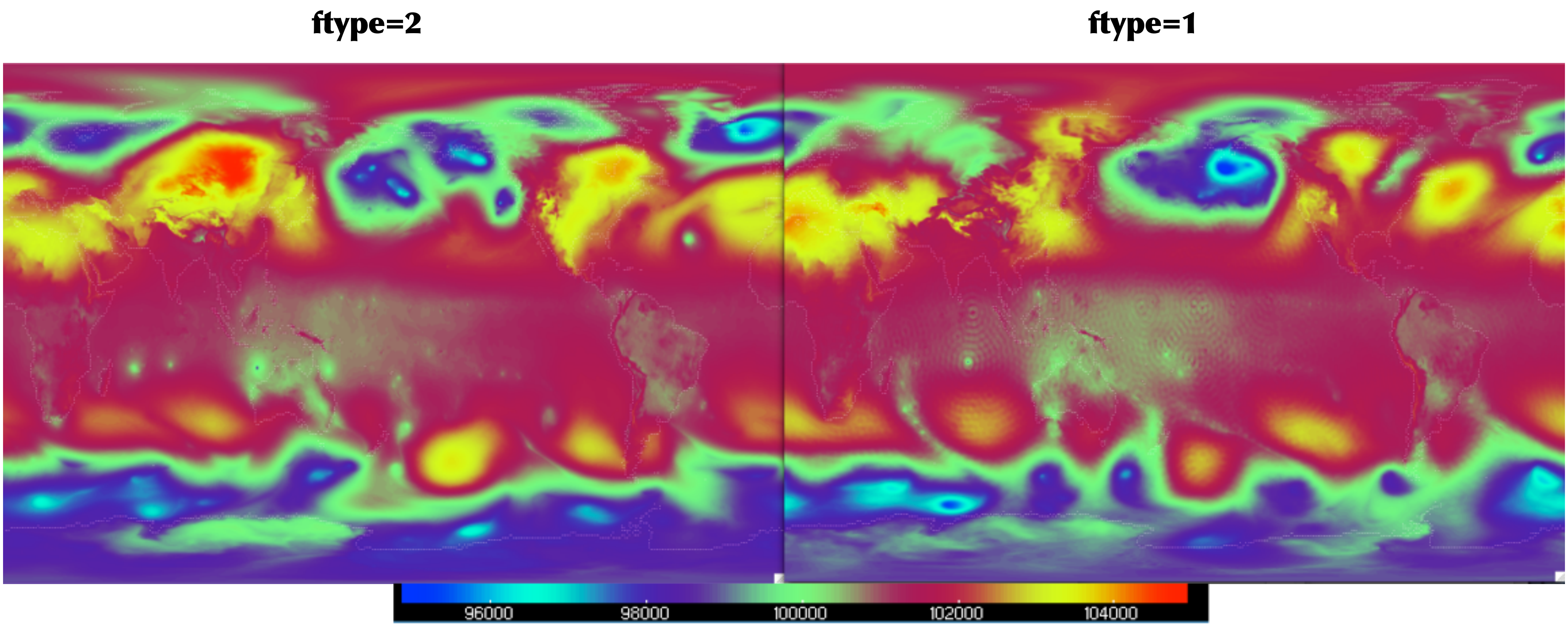
Physics dynamics coupling methods in CAM-SE: se_ftype=2

Advance dynamics core (30 minutes): for (u,v,T) add physics tendency "chunks" during the dynamics time-stepping
- every 15 minutes in this example
(I refer to it as "dribbling")

Compute physics tendencies based on dynamics updated state

Split physics tendencies into a number of "chunks" for u,v,T
Update tracer state with physics tendencies

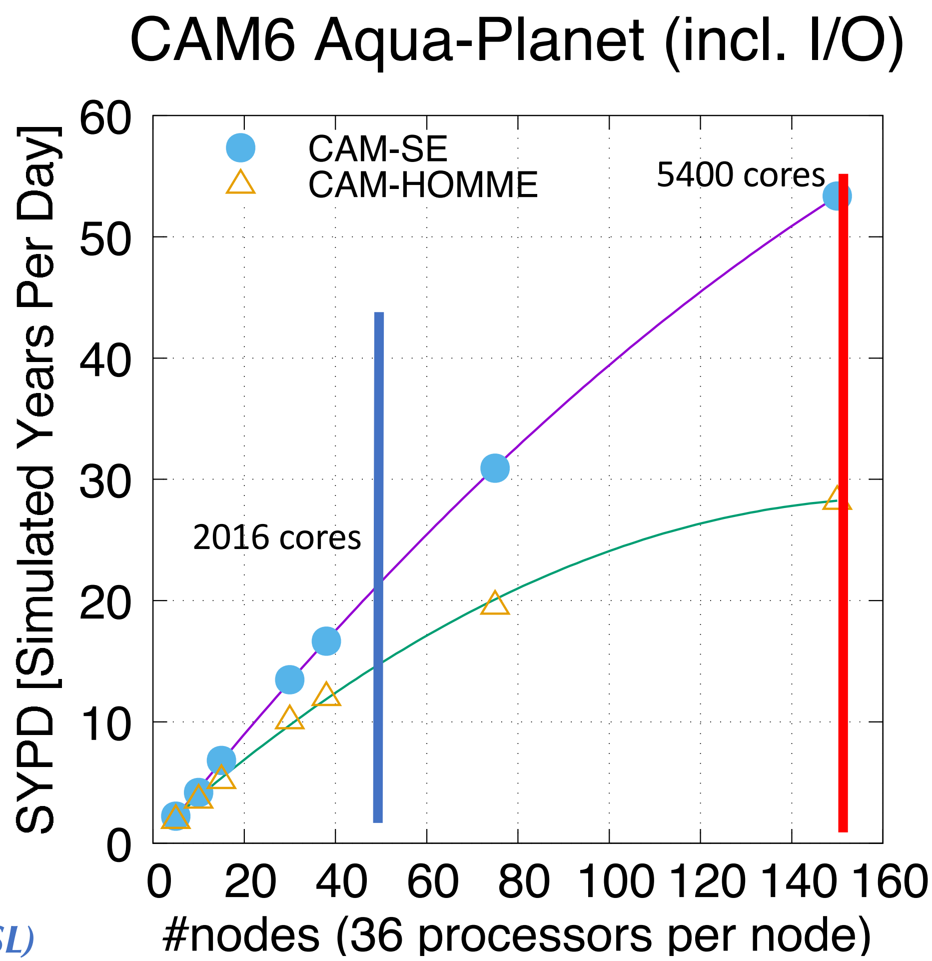
Instantaneous PSL for CAM-SE at approximately ¼ degree horizontal resolution



Outline

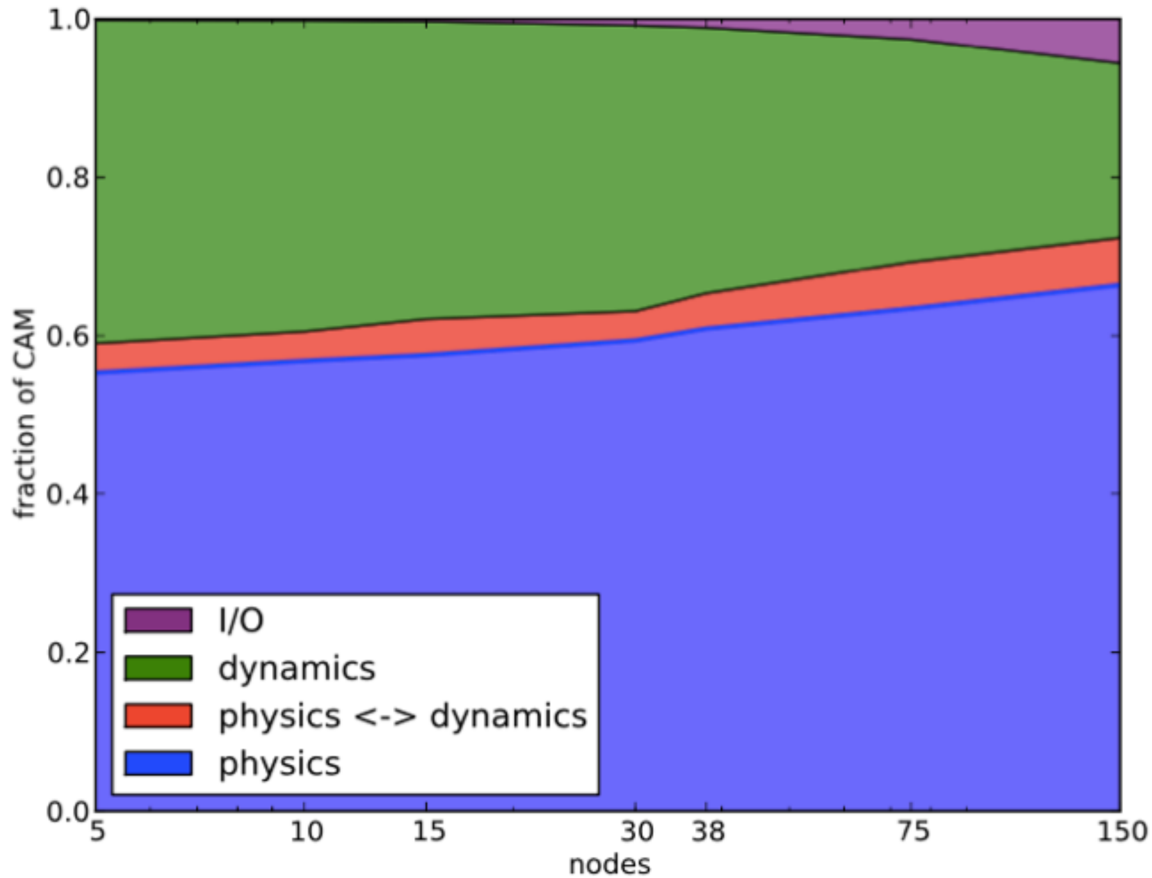
- Previous talk: derivation of equations of motion …
- Introduce computational grid and projections
- Introduction to the spectral-element method
- Properties of the spectral-element
- Time-stepping
- Coupling to physics
- **Computational performance**
- Variable resolution version of CAM-SE

Computation throughput (ne30 ~ 1 degree)



Data produced by John Dennis (CISL)

Computation throughput (ne30 ~ 1 degree)



Data produced by John Dennis (CISL)

Outline

- Previous talk: derivation of equations of motion …
- Introduce computational grid and projections
- Introduction to the spectral-element method
- Properties of the spectral-element
- Time-stepping
- Coupling to physics
- Computational performance
- **Variable resolution version of CAM-SE**

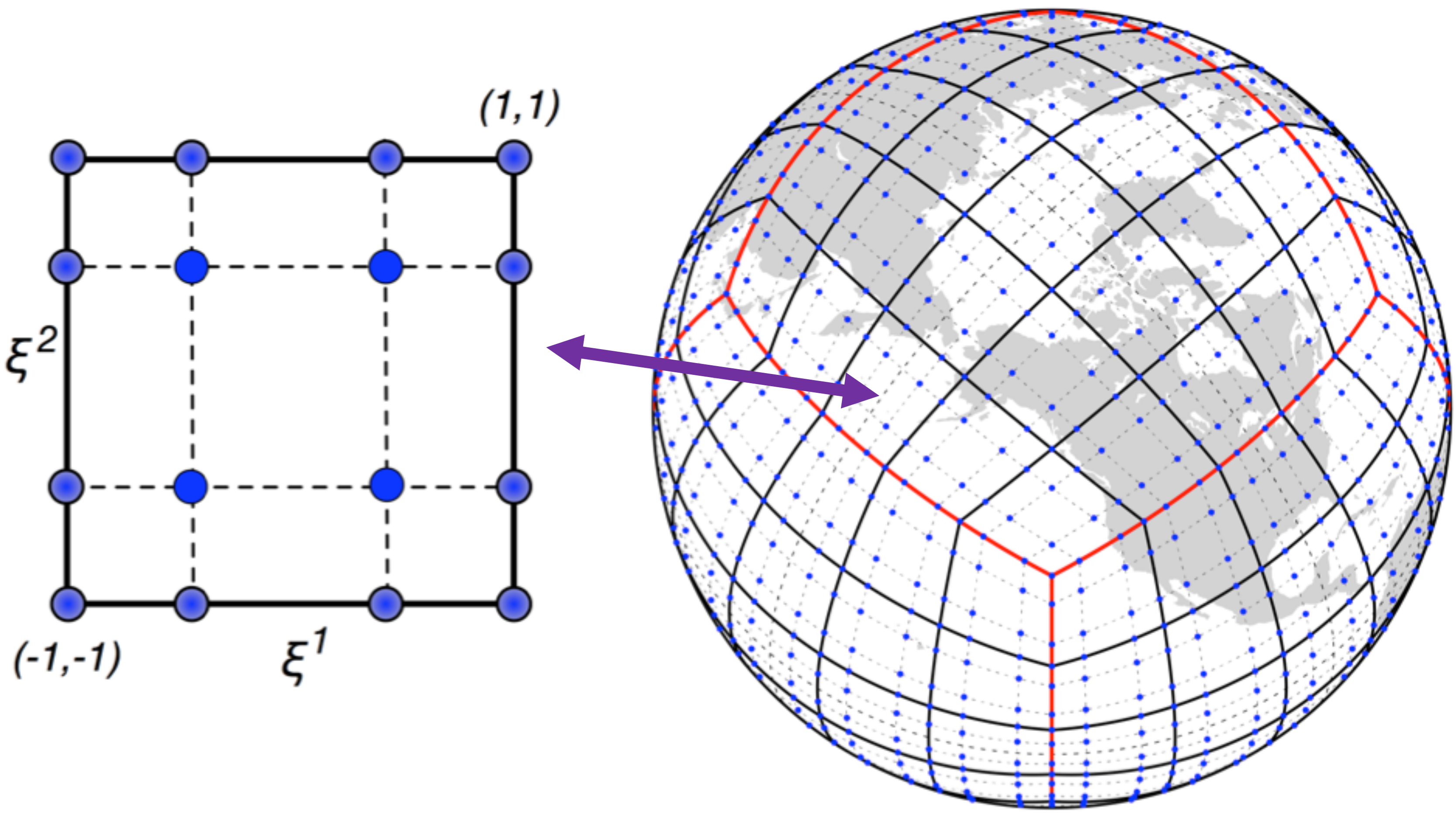
Variable horizontal resolution

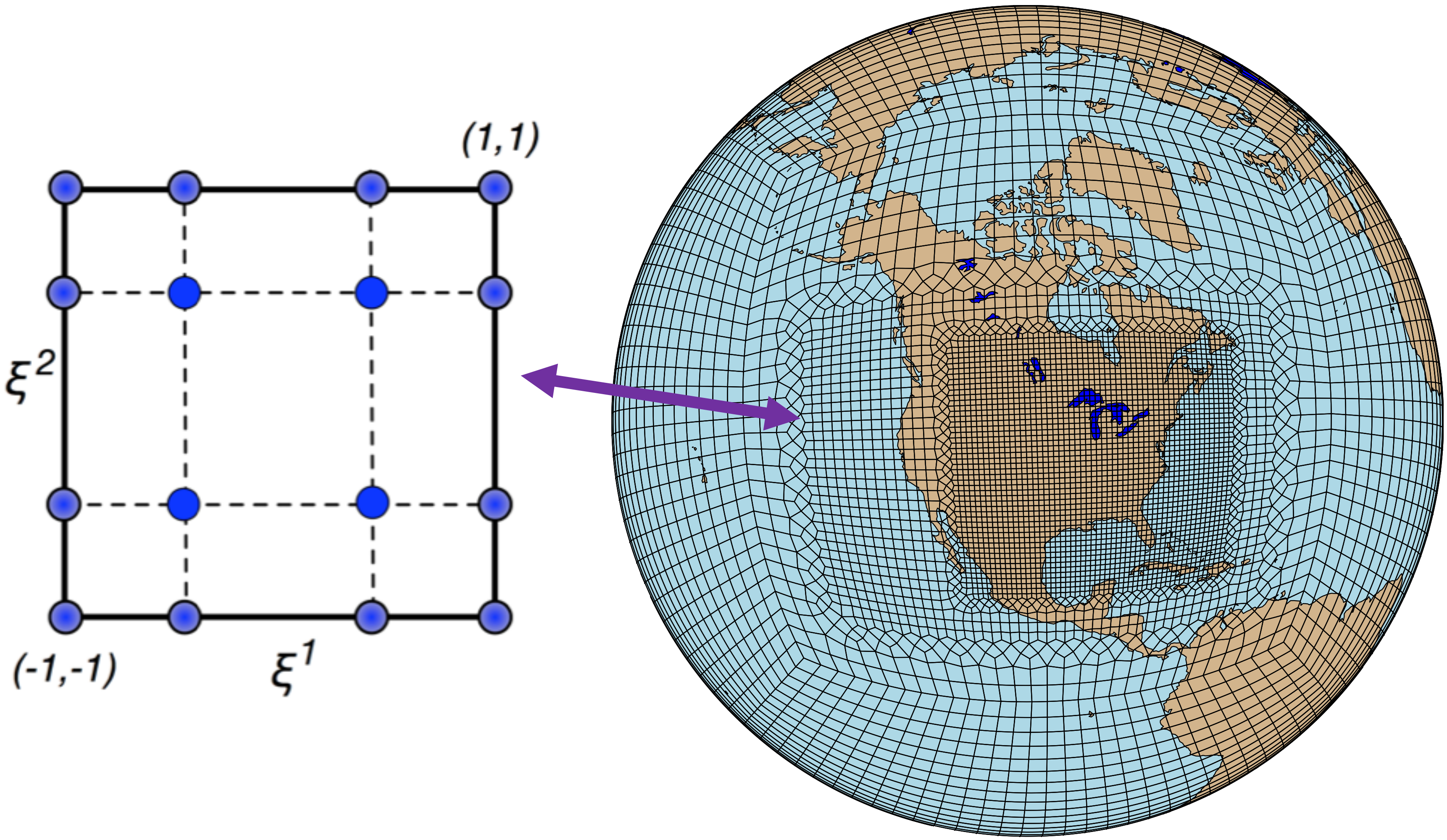
- **One of the advantages of the spectral-element method is that it can relatively easily be adapted to variable resolution meshes (as long as elements are quadrilateral)**

A direct way to address this problem is establishing a transformation $\mathcal{J}_e : \Omega_e \rightarrow [-1, 1]^2$, where $\mathcal{J}_e$ may be considered as a composite mapping combining the gnomonic and the quadrilateral to standard-element mapping. Let the Jacobian associated with the composite mapping be $J_e = J_e(\sqrt{G})$. Then an arbitrary surface integral on Ω_e can be expressed in terms of local coordinates $\xi^1, \xi^2 \in [-1, 1]$ and the Jacobian J_e

$$\int_{\Omega_e} \psi(x^1, x^2)\, d\Omega_e = \int_{-1}^{1}\int_{-1}^{1} J_e(\xi^1, \xi^2)\, \psi(\xi^1, \xi^2)\, d\xi^1\, d\xi^2 \approx \sum_{k=0}^{N}\sum_{l=0}^{N} w_k w_l\, J_e(\xi_k^1, \xi_l^2)\, \psi(\xi_k^1, \xi_l^2), \tag{80}$$

where w_k, w_l are the Gauss quadrature weights.


(1,1)
ξ^2
(-1,-1)
ξ^1

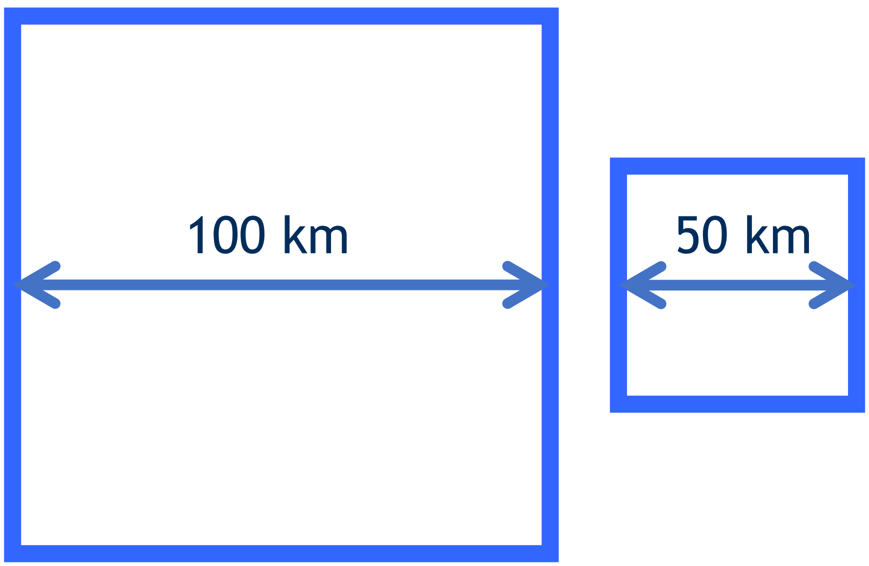


(1,1)
ξ2
(-1,-1)
ξ1

Challenges: diffusion

CAM-SE, Zarzycki et al., 2014, JClim

$$K_4\left(\Delta x\right) = K_4\left(\Delta x_{ref}\right)\left(\frac{\Delta x}{\Delta x_{ref}}\right)^{y}$$

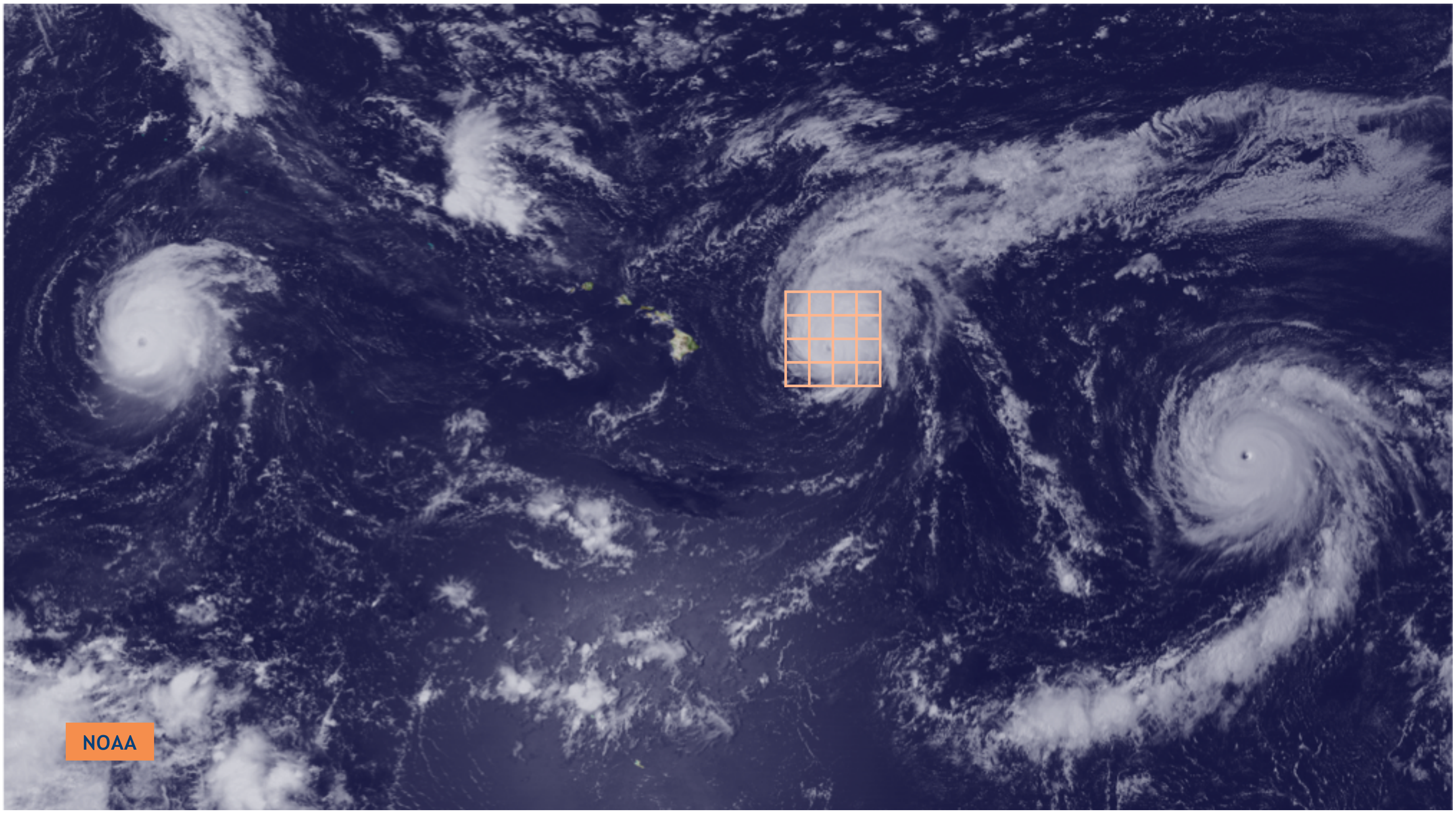


Let's say 100 km cell is our "reference" and y = 3.321

$$\left(\frac{50}{100}\right)^{3.321} = \frac{1}{10}$$

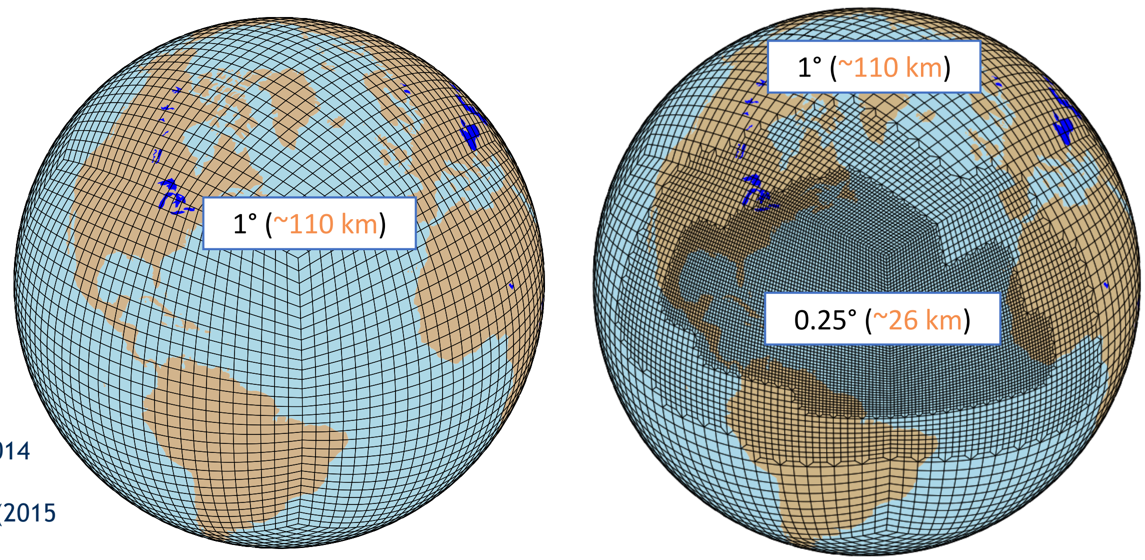
Our diffusion coefficient (K_4) in the 50 km box is 1/10th that of 100 km box!

V-R applications: tropical cyclones



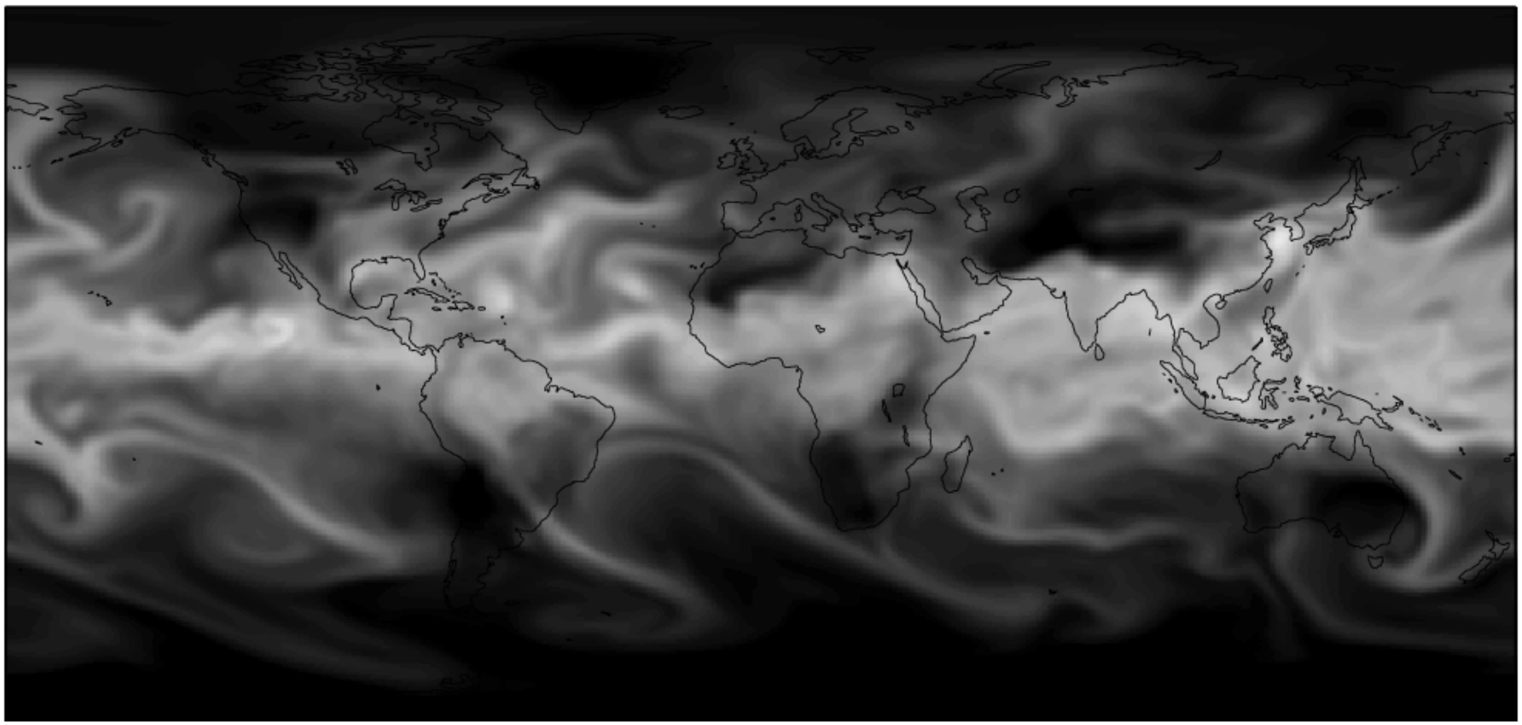
V-R applications: tropical cyclones

- Community Atmosphere Model Spectral Element (CAM-SE)
- Atmospheric Model Intercomparison Project **(AMIP)** protocols
  - 1980-2002 (23 years)
    - Prescribed SSTs, ozone, aerosols, solar insolation
- Simulate historic, observed climate



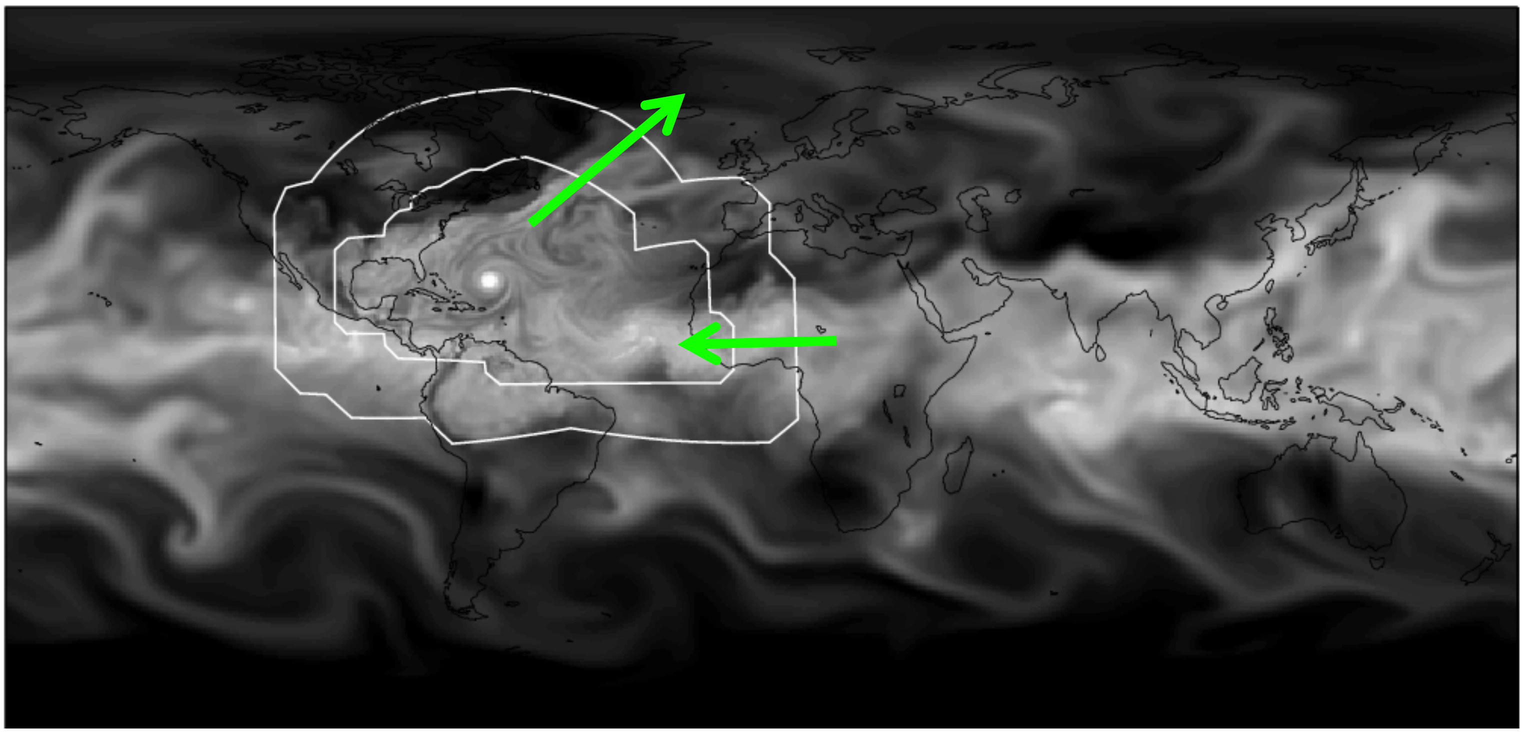
Zarzycki and Jablonowski, (2014 JAMES)
Zarzycki et al., (2015 JClim)

Uniform global simulation



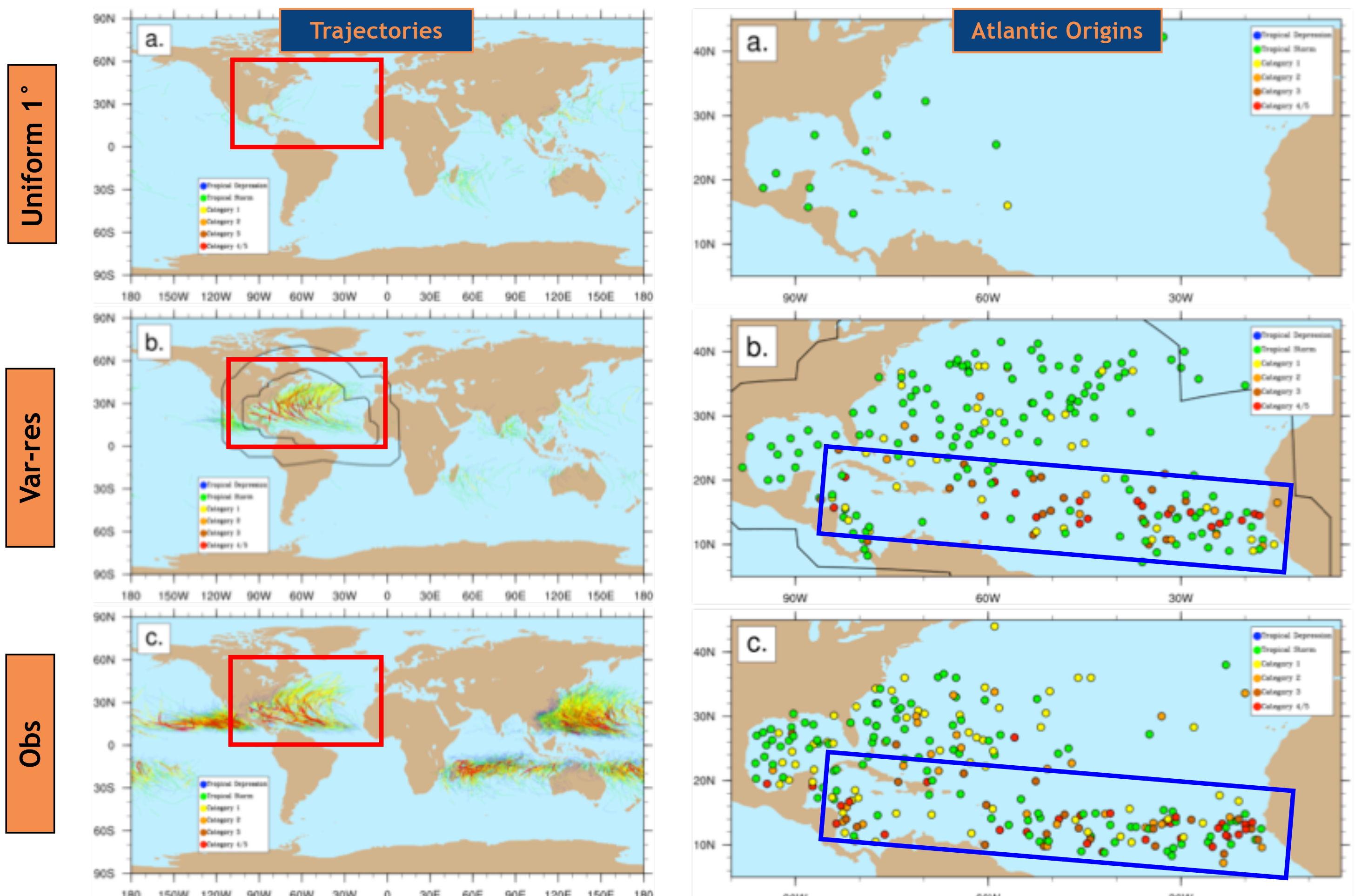
Precipitable water, Sept 1-16

Variable-resolution global circulation



Precipitable water, Sept 1-16

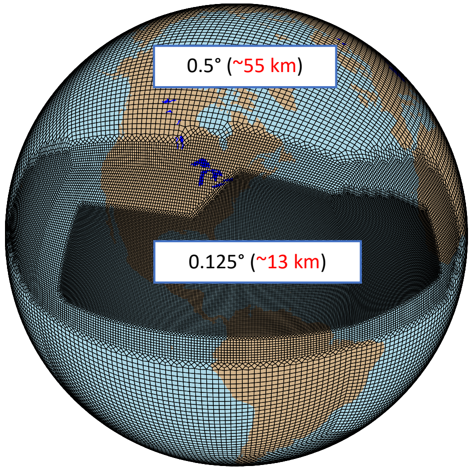
Tropical cyclones



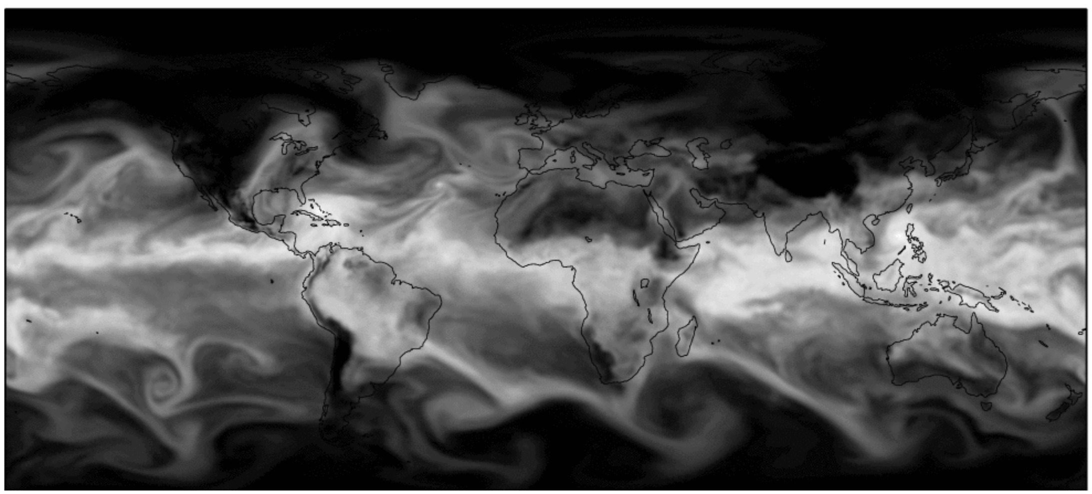
Zarzycki and Jablonowski, 2014, JAMES

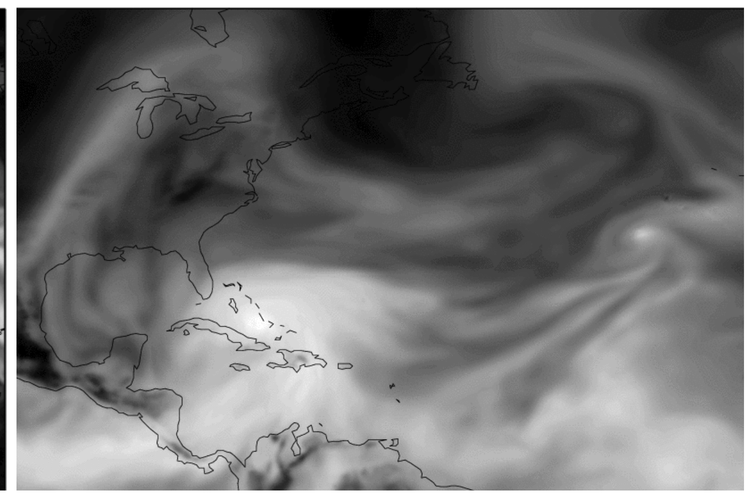
Numerical weather prediction

- 8 day forecast = ~1.5 hours of wall clock time on 800 cores (NCAR Yellowstone)
  - **~6-7x cheaper than a globally-uniform 13 km forecast**



Sandy TPW:
INIT 12Z
10/25/12





Outline

- Previous talk: derivation of equations of motion …
- Introduce computational grid and projections
- Introduction to the spectral-element method
- Properties of the spectral-element
- Time-stepping
- Coupling to physics
- Computational performance
- Variable resolution version of CAM-SE

More information: http://www.cgd.ucar.edu/cms/pel
Email: pel@ucar.edu