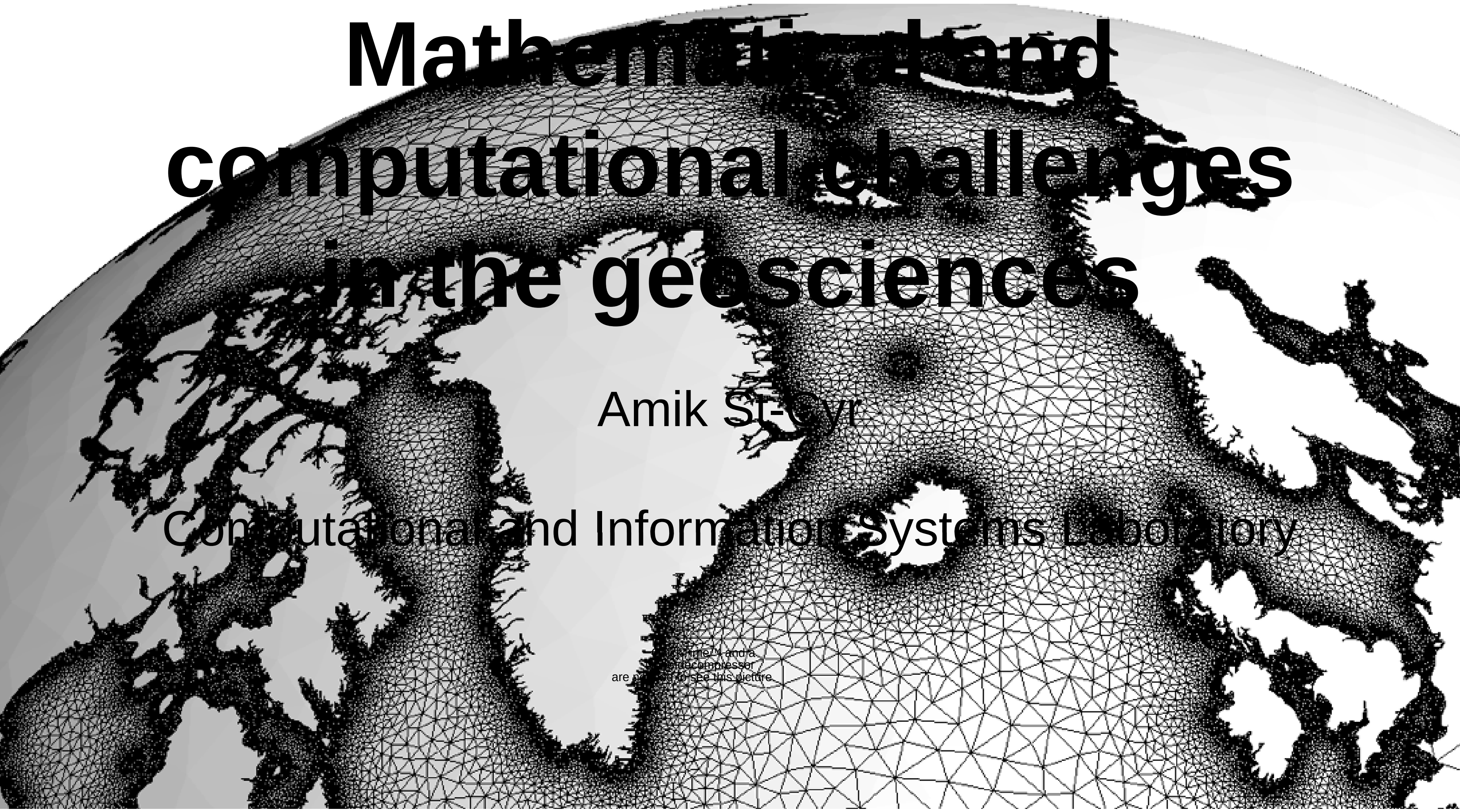
Mathematical and computational challenges in the geosciences

Amik St-Cyr

Computational and Information Systems Laboratory



Contributors to this talk

- Adrian Sandu VTU
- David Neckels (formerly NCAR)
- Dimitri Mavriplis UW
- Mark Taylor Sandia
- J-F Remacle (Louvain la Neuve)

NCAR's computational mathematics group

- Development of novel numerical methodologies for geosciences
- Background: NWP, Applied mathematics, HPC and CFD
- High-order methods for PDEs: CG, DG, RBFs
- FVM, AMR, Preconditioning for HOM
- Mesh-less methods
- Time-stepping procedures, parallel algorithms (HPC)
- Coronal Mass ejection, climate modeling, weather: prototyping
- Piotr Smolarkiewicz, Natasha Flyer, Ram Nair and Amik St-Cyr
- Experience on: daily O(1000) processors, HOMME only geo-code(?) scaling to O(100k) ...

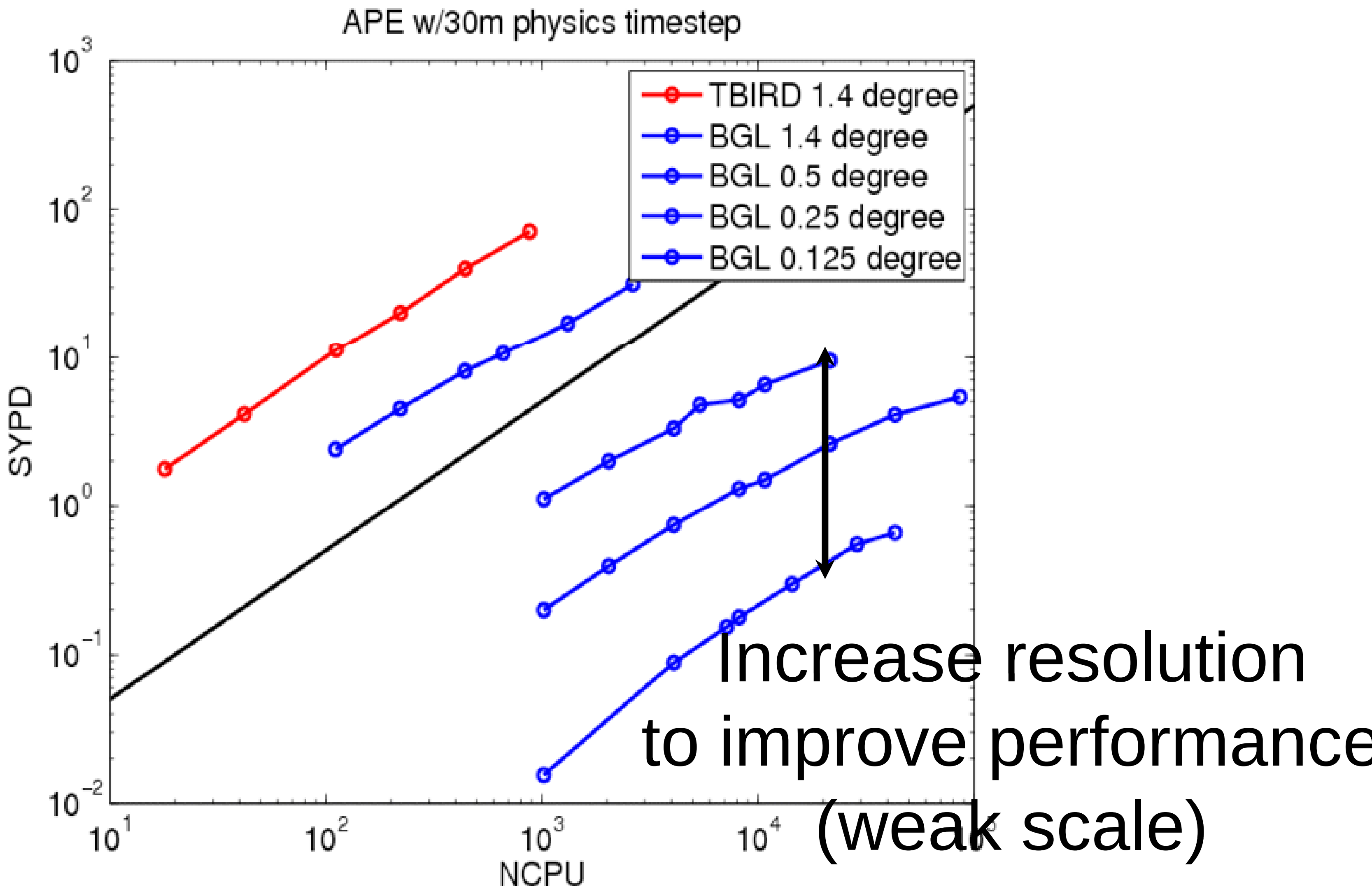
Outline

- What challenges?
- Algorithms involved
- Example on how to tackle challenges
- Conclusions?

What challenge?

- O(1 Million) processors:
  - How to use them efficiently?
  - Is this for specialists only? Is there an “in-between”?
  - Current codes only strong scale!
- Everyone in the geo has its own code: ← “Monthly” climate!
  - Over spending in software engineering resources...
  - Costly re-writes to take advantages of latest ideas
  - Solves PDEs ????
- What’s the “ultimate” goal ? ← Why rewrite a new code EVERY time?
  - Give the best possible answer to the PDE’s given the computational resources
  - Optimal algorithms: O(N)

O(100k)



- HOMME code: aquaplanet with CCSM

Multiplication of codes

Mostly:

- Solution of the equation of motion (PDEs): climate modeling, weather prediction, MHD, seismic, coronal mass ejection, Ocean modeling ...

Also:

- Solution of chemistry, physics equations: stiff ordinary differential equations (ODEs)

Multiplication of codes

- Most of us (try) to solve:

$$\mathbf{U}_t = -\nabla \cdot \mathbf{F}(\mathbf{U}, \nabla \mathbf{U}) + \sum_{i=1}^{N} S_i(\mathbf{U}, \nabla \mathbf{U}, t) =: H(\mathbf{U}),$$

$$G(\mathbf{U}, \nabla \mathbf{U}) = 0.$$

- All of continuum mechanics...
- Evaluate only rhs of PDE: $H_h(\underline{\mathbf{U}}_h)$
- Explicit, Semi-Implicit, FI, Jacobian Free...
- Scientist could concentrate on rhs
- Control of h enables AMR

Goal?

QuickTime™ and a
YUV420 codec decompressor
are needed to see this picture.

0.3125 degrees...

$$|\zeta| \geq 3 \times 10^{-5}\ \mathbf{s}^{-1}$$

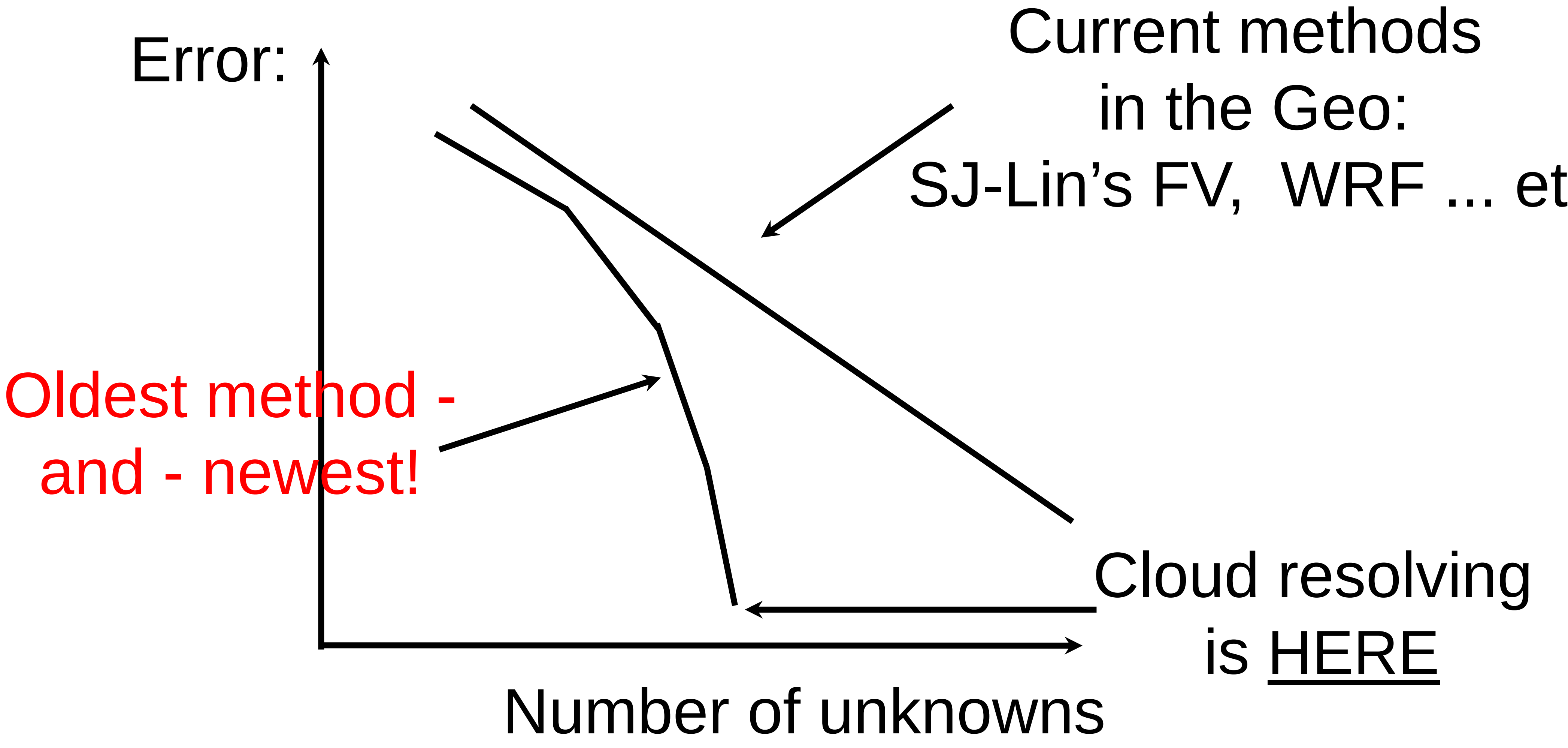
Goal?

- Given “P” processors compute best solution in optimal number of operations
  - Control of the error (when available!) is paramount!
  - If not??
    - Use adjoint + AMR (node movement)
    - Equi-distribute error: any other functional works
    - Galerkin ...

Algorithms in geo...

- Time-integration: explicit, split-explicit, semi-implicit, implicit, LMM, RK, Multi-rate, IMEX, exponential integrators...
- Space integration: SEM, DGM, FDM, FVM
- Elliptic problems: direct methods, iterative methods: KSP, multi-grid, preconditioning ...
- Optimization techniques

High vs low order?



High vs low order?

- High-order bad for under-resolved...
  - We <u>need h-p</u> (use low order methods only where necessary...)
- High-order in time or implicitness <u>does help</u>

QuickTime™ and a
decompressor
are needed to see this picture.

QuickTime™ and a
decompressor
are needed to see this picture.

Agenda:

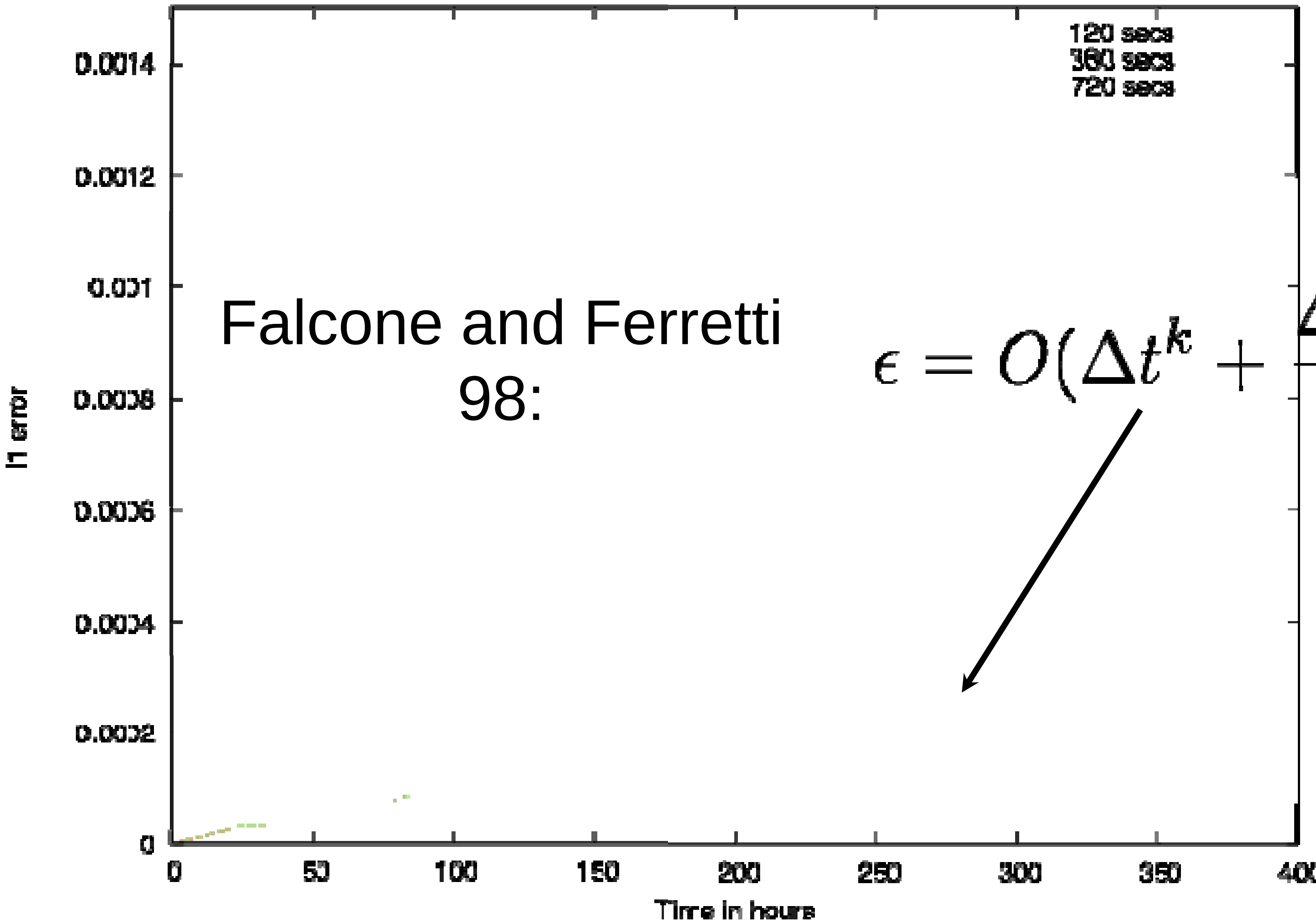
- Mathematically sound framework
- Generic enough to solve more than one problem
- Unstructured grids (contains structured!)
- Error estimation: adjoints with equi-distrib (best solution possible given computational resources)
- “All”-orders: hides cost of unstructured
- Advanced time-stepping: parareal(?), multi-rate, multi-methods, linearly implicit...?
- ... Capable of weak scaling our problems low communication costs/halos ...

Piece of CAKE!

SISL

- Semi-Implicit + Semi-Lagrangian
- Gravity waves and advective time-scale
- Proposed by A. Robert (81)
- Parallel issues in its classical version...
- Use idea of Maday et al. (90)
- N-L version for sw: (A and Thomas 05)
- Acceleration is 4 wrt explicit version
- Problem solved! ...

Comparison with reference solution from NCAR pseudo spectral core



Falcone and Ferretti 98:

$$\epsilon = O(\Delta t^k + \frac{\Delta x^p}{\Delta t})$$

Fully implicit

- DG in space for Euler equations: WRF form
- New Rosenbrock W-method
- No non-linear cycles (Newton)
- No Jacobians: Jacobian free
- Low Mach preconditioning
- Element block Jacobi
- Results on benchmark tests
- Acceleration: 3 to 45 wrt to explicit version

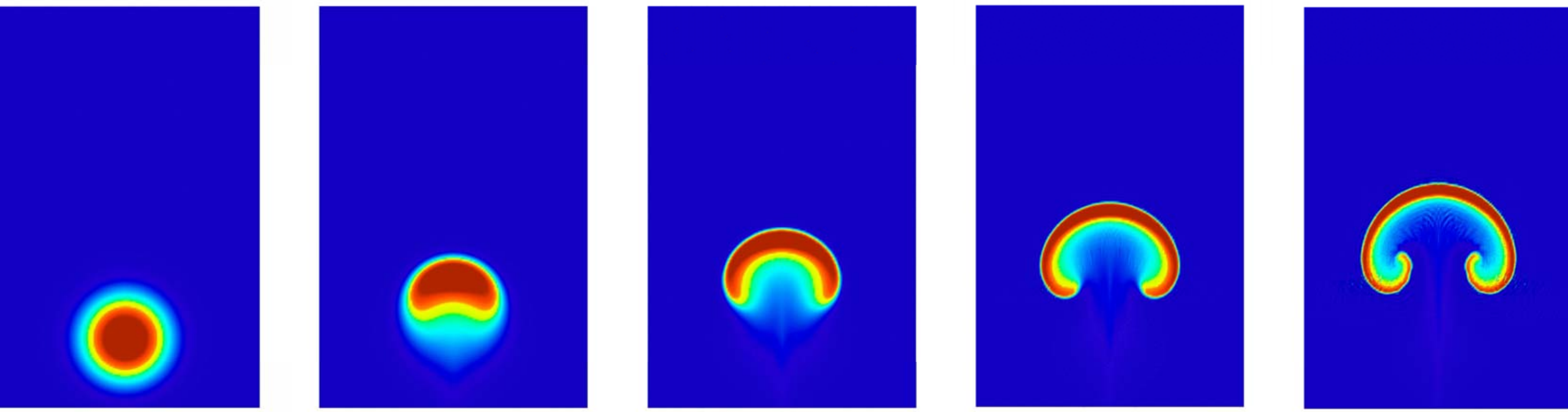
Effects of low Mach

solver tolerance ~ 1E-6, (Nx,Nz)=(16,8), p=7, 180 meters resolution (approx.)

time step	W LM	accel	WO LM	accel
1.0s	30	3.2	33	2.8
2.0s	36	5.1	45	4.1
10.0s	69	13.5	103	9.1
50.0s	207	22.7	493	10.2

"Wicker" Bubble: Wicker and Skamarock MWR02

Rising bubble

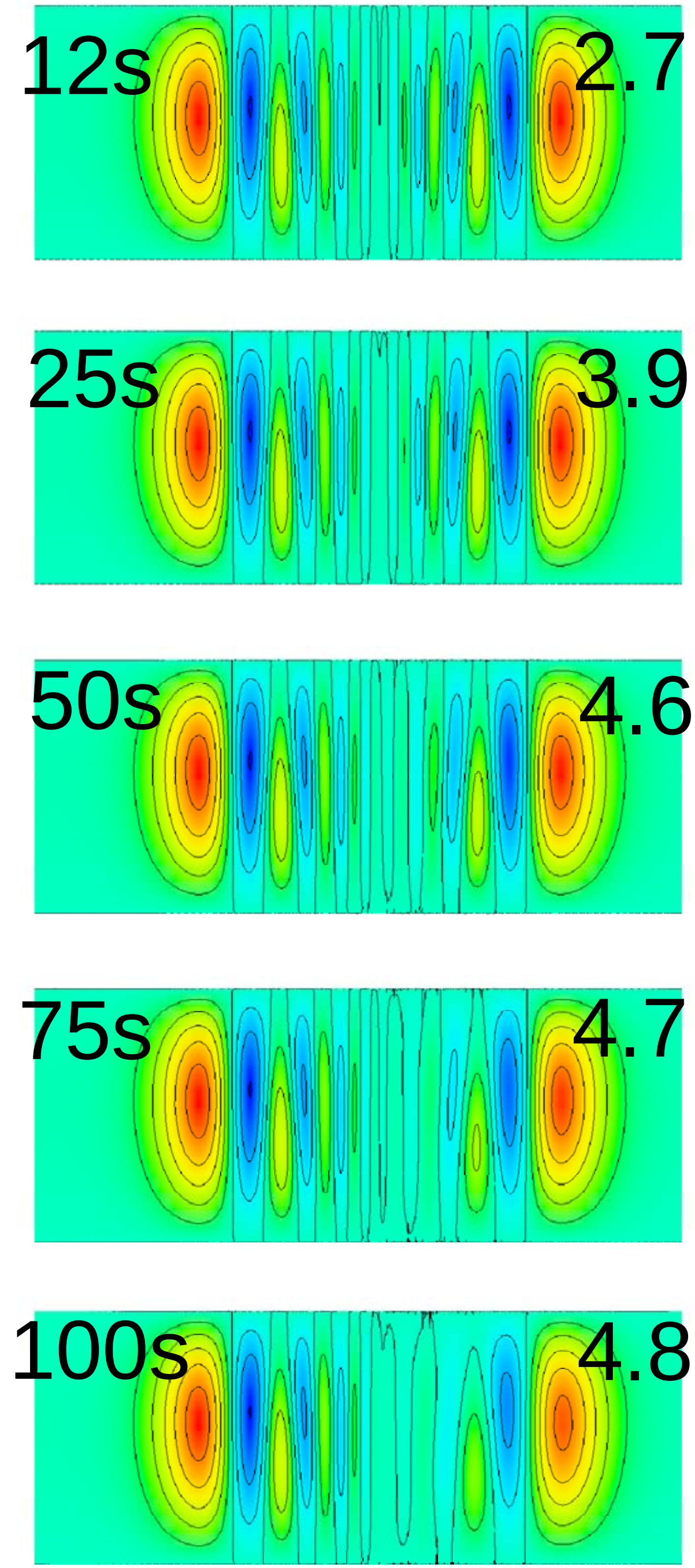


5 meters resolution, p =7, Tf = 600 secs

Inertia Gravity

- Inertia gravity wave in channel + bg flow
- dx=dz=500m, poly order 8, nez=3, nex=90
- dt=12, 25, 50, 75, 100 seconds
- Accelerations: 2.7, 3.9, 4.6, 4.7, 4.8 wrt explicit
- 20 m/s to the right

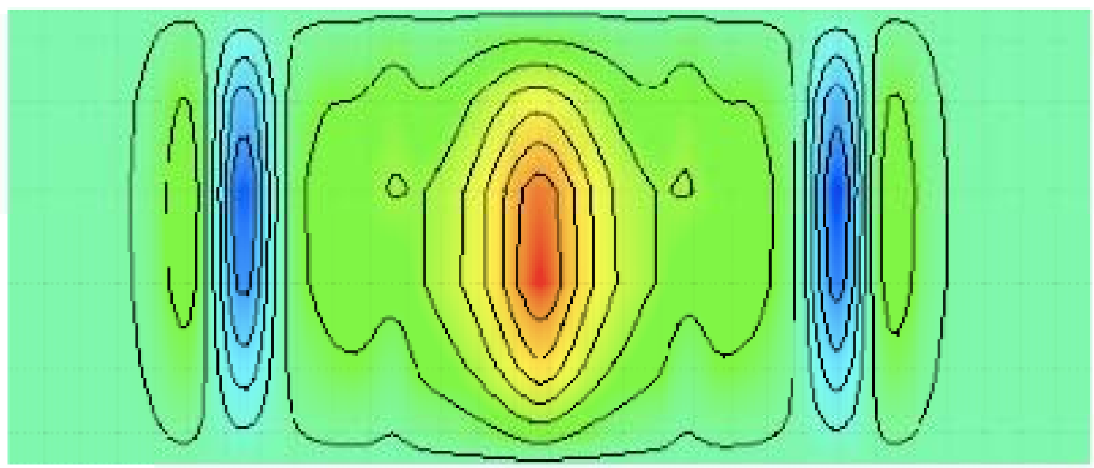
Skamarock and Klemp 02



Inertia Gravity wave

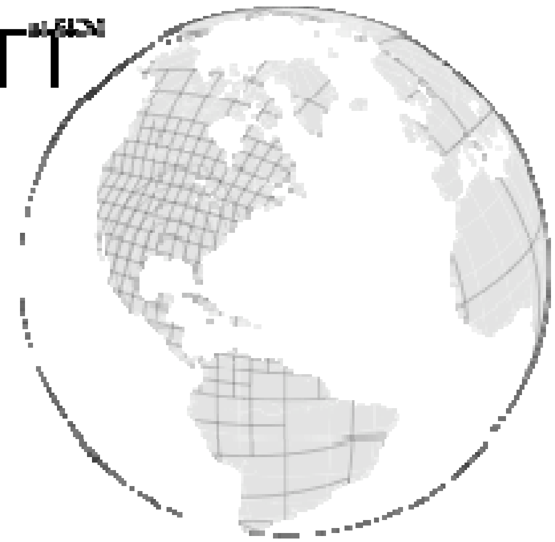
- Eady model (one more equation + Coriolis)
- Very thin channel (hydrostatic: shallow atm)
- 1 element in the vertical
- 600 in the horizontal (1km x 1km resolution)
- p=7
- accel > 45

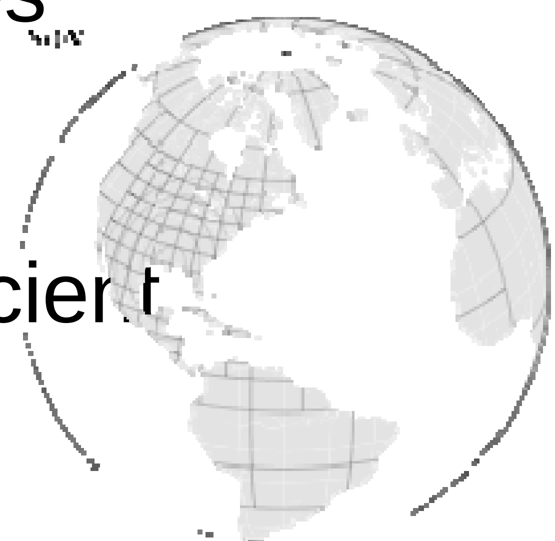
Skamarock and Klemp 02



AMR

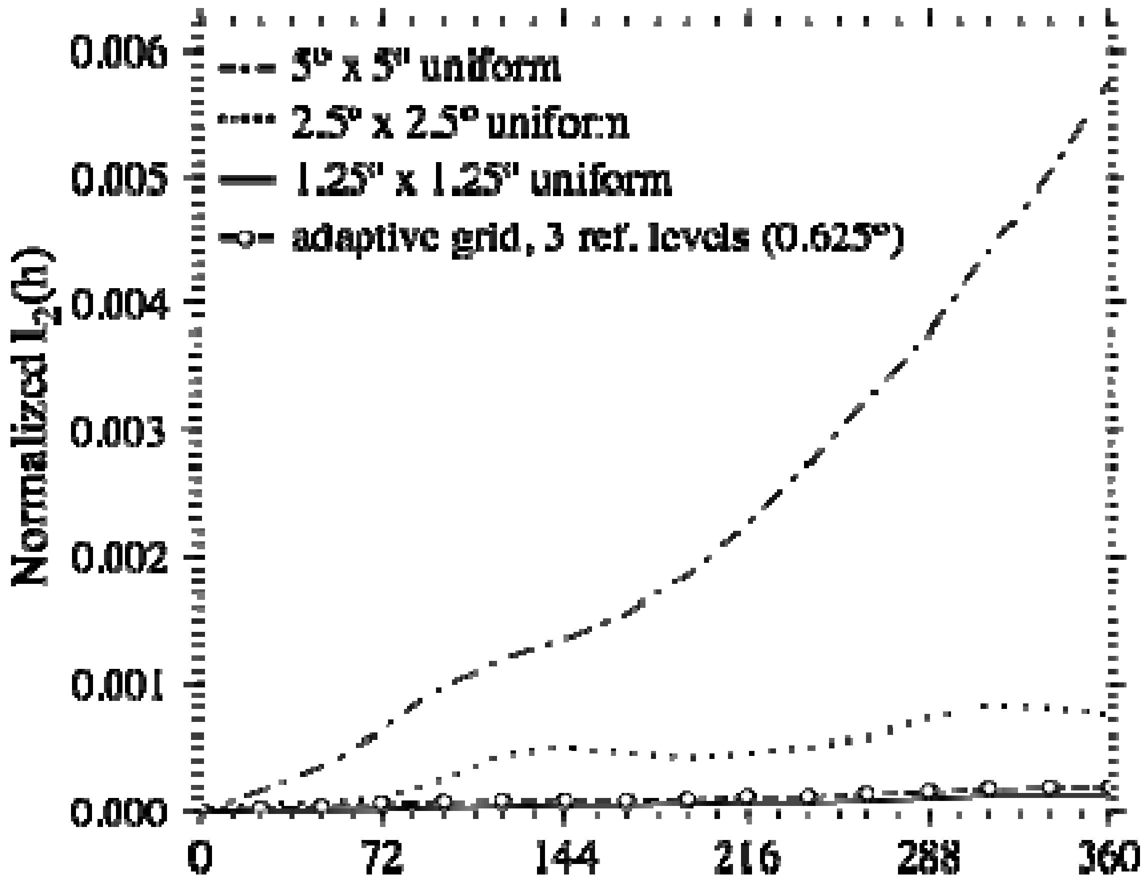
- Comparison of SEM with FVM (SJDTT 08)
- Both non-conforming dynamic approaches
- Halos issues for FVM: error increases
- Cubed sphere (SEM) lat-lon (FVM)
- At comparable errors SEM more efficient
- Runs below 1/3 degrees on 16 processors!





Flow impinging a mountain

High-resolution solution DWD (German weather service) T426



SEM error 10 times lower than FVM

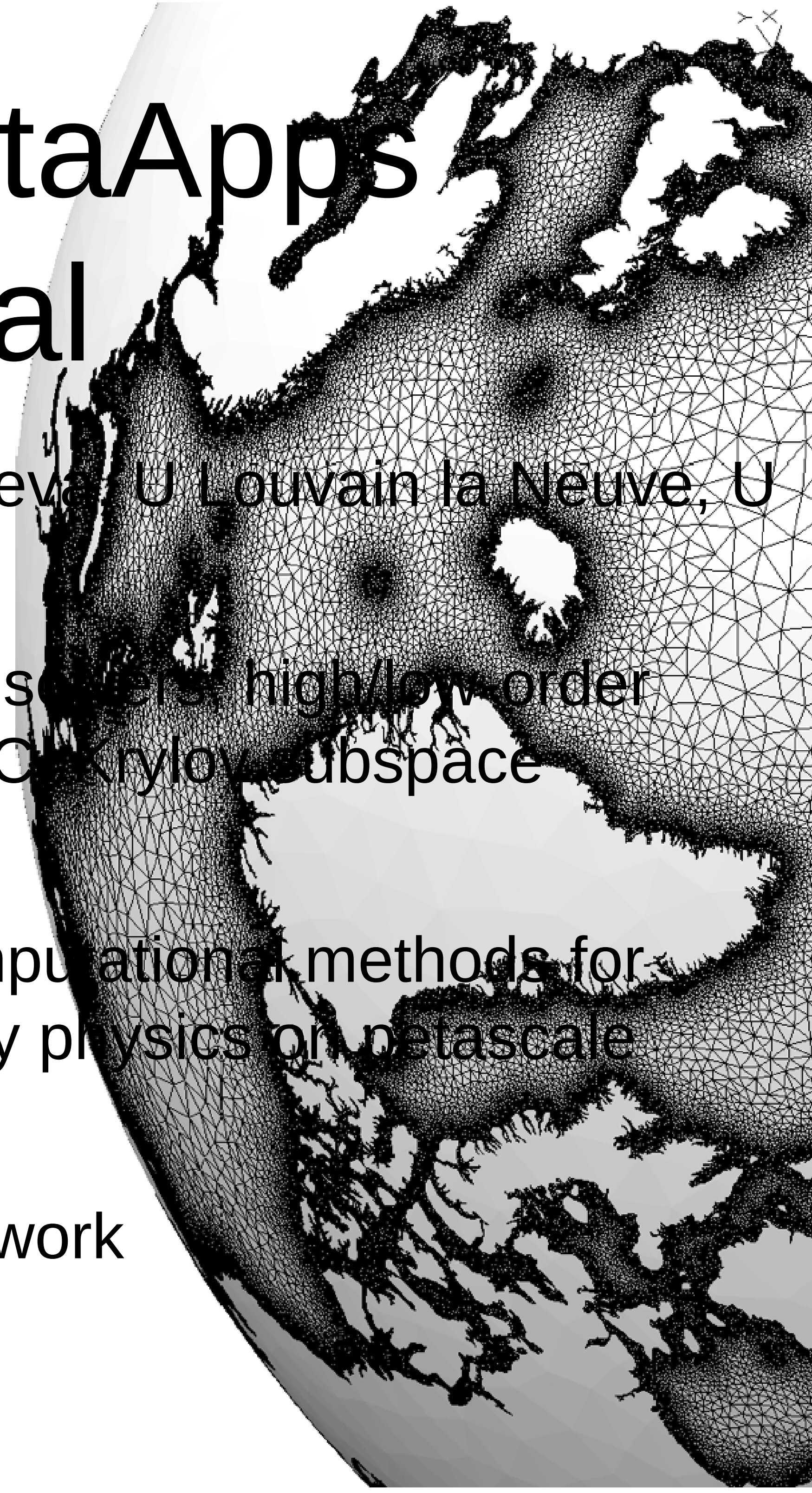
To generate same error one more ref in FVM: 6.5 slower than SEM

NSF-CDI/PetaApps proposal

- Multi-institutional: VTU, UW, U Geneva, U Louvain la Neuve, U Nice Sophia-Antipolis
- Expertise in: time-stepping, optimal solvers, high/low-order methods, software engineering, HPC, Krylov subspace methods, adjoints.

Text

- Goal: The discovery of efficient computational methods for multiscale adaptive, multidisciplinary physics on petascale system
- Build an all scales simulation framework



Thank you!:

- Mathematically sound framework
- Generic enough to solve more than one problem
- Unstructured grids (contains structured!)
- Error estimation: adjoints with equi-distrib (best solution possible given computational resources)
- "All"-orders: hides cost of unstructured
- Advanced time-stepping: parareal(?), multi-rate, multi-methods, linearly implicit...?
- ... Capable of weak scaling our problems low communication costs/halos ...