Scalability of Elliptic Solvers in Numerical Weather and Climate- Prediction

Eike Hermann Müller, Robert Scheichl

Department of Mathematical Sciences, University of Bath

ECMWF Workshop on the Use of HPC in Meteorology

Reading Oct 1st, 2012

UNIVERSITY OF BATH

NGWCP project

Next **G**eneration **W**eather and **C**limate **P**rediction project

- *Selection of numerical algorithms to simulate the atmosphere in weather and climate prediction which take advantage of **massively parallel** architectures.*
- Develop new **dynamical core** for the Met Office Unified Model which scales up to $10^5 - 10^6$ cores
- Substantial **increase in global model resolution**

$$\sim 25\text{km} \quad \rightarrow \quad \sim \text{few km}$$

$$\Rightarrow \gtrsim 10^{10} \text{ degrees of freedom per atmospheric variable}$$

- Model runtime $\lesssim$ 1hour for 5 day forecast
- Solve **elliptic PDE** for pressure correction in $\ll$ 1second

1. Background
   - Elliptic PDE in implicit time stepping
   - Model equation
   - Multigrid solvers
2. Scaling results
   - Massively parallel scaling on `Hector`
3. Tensor product geometric multigrid
   - Parallel scaling results
     - Weak scaling
     - Strong scaling
   - Implementation in DUNE-Grid
4. Summary and Outlook

Implicit timestepping

Large scale atmospheric flow:

Navier Stokes equations

$$\frac{D\boldsymbol{u}}{Dt} = -2\boldsymbol{\Omega}\times\boldsymbol{u} - \frac{1}{\rho}\nabla p + \boldsymbol{g} + \boldsymbol{S}^{\boldsymbol{u}}$$

$$\frac{D\rho}{Dt} = -\rho\nabla\cdot\boldsymbol{u}, \qquad \ldots$$

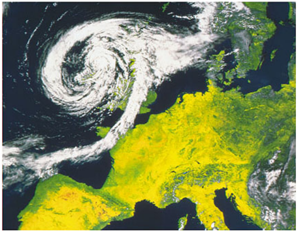


image source: NASA

Implicit time stepping

- Unconditionally stable ⇒ Larger integration time step Δt
- Solve 3d **elliptic PDE** for pressure correction π' at every time step [Davies et al. Q J Royal Met Soc, 131 (608):1759-1782, 2005, . . .]

$$-(\alpha\Delta t)^2 c_s^2 \nabla\cdot(a\nabla\pi') + b\pi' = RHS$$

- Significant proportion of model runtime
- Need **numerically efficient & scalable** solver

Does the solver scale?

Started by testing the following "black box" solvers:

Distributed and Unified Numerics Environment (DUNE)

ISTL Bastian et al. 2008, Blatt and Bastian 2007 & 2008

- CG preconditioned with aggregation AMG + ILU0 smoother

Hypre Developed at LLNL by U. Maier-Yang, R. Falgout and others

- CG preconditioned with BoomerAMG

Matrix (+ AMG) setup costs?

$\Rightarrow$ **"Matrix-free" geometric multigrid**

- Hand-written Fortran code based on tensor-product multigrid idea Börm, Hiptmair 2001. Numerical Algorithms. 26: 219234
- DUNE-based code with indirect horizontal-, direct vertical-addressing

Does the solver scale?

Comparison of **Multigrid solvers** for model equation

Weak scaling of # iter, total time +AMG setup time

all times in seconds

# proc	# dof	AMG (DUNE)		BoomerAMG		**geo MG**	
16	$8.3 \cdot 10^6$	11	6.92+4.13	12	8.72+2.59	6	**1.99**
64	$3.4 \cdot 10^7$	11	7.01+4.92	13	9.52+2.74	6	**2.02**
256	$1.3 \cdot 10^8$	11	7.18+4.88	12	8.98+2.82	6	**2.04**
1024	$5.4 \cdot 10^8$	11	7.32+5.89	12	9.04+3.18	6	**2.06**
4096	$2.1 \cdot 10^9$	13	8.64+6.32	12	8.99+3.56	6	**2.06**
16384	$8.6 \cdot 10^9$	12	8.16+8.06	11	9.43+5.75	6	**2.10**
65536	$3.4 \cdot 10^{10}$	11	7.49+10.92	9	20.20+7.09	6	**2.24**

+ matrix setup time for AMG solvers

Model equation

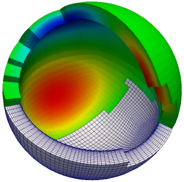
Simplified model equation for $u \equiv \pi'$ on spherical shell

$$-\omega^2 \left[\Delta_{(2d)} + \lambda^2 \frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2 \frac{\partial}{\partial r}\right)\right] u + u \quad = \quad RHS$$

Dimensional analysis: $r \in [1, 1+h]$ with $h = H/R_{\text{earth}} = 10^{-2}$:

$$\omega^2 \sim \left(\frac{c_s \alpha \Delta t}{R_{\text{earth}}}\right)^2 \qquad \lambda^2 \sim \frac{1}{1 + (\alpha \Delta t)^2 \left(N^{*0}\right)^2}$$

- Acoustic waves: $c_s \approx 550 ms^{-1}$
- Buoyancy frequency $N^{*0} = 0.018 s^{-1}$
- Off-centering parameter $\alpha = \frac{1}{2}$
  (fully implicit: $\alpha = 1$, fully explicit: $\alpha = 0$)



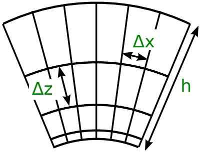
Model equation

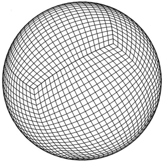
Properties

- $h = H/R_{\text{earth}} \approx 1/100 \Rightarrow \lambda^2/h^2 \gg 1$
- Strong **vertical anisotropy** $\left(\lambda/h \cdot \frac{\Delta x}{\Delta z}\right)^2$
- **Constant term** improves condition number (on coarser MG levels)

$$-\omega^2 D^{(2)} u + u = RHS$$

- Horizontal grid e.g. cubed sphere, icosahedral,...
  no pole singularity as in lat/lon grid





Multigrid solvers

Multigrid idea:

Eliminate error on all scales

- Hierachy of grids $h, 2h, 4h, \ldots$
- Apply smoother (e.g. SOR) on all levels, restrict/prolongate between levels
- Residual equation on coarser grids

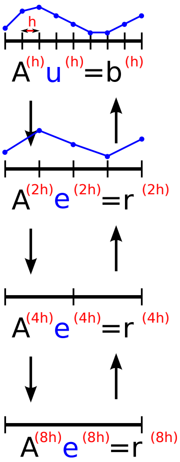
$$A^{(H)} e^{(H)} = r^{(H)}$$

$\Rightarrow$ **Work on coarse grids is cheap!**

- Algorithmically optimal

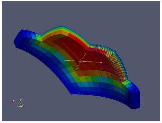
$$Cost(MG) = \mathcal{O}(n)$$

- Robust & parallelisable



Setup

Weak scaling



- 1/6 of cubed sphere grid
  (have also run on entire sphere)
- Horizontal partitioning only* (atmos. physics)
- # processors $\propto$ problem size

$$n_x \mapsto 2n_x, \qquad n_y \mapsto 2n_y, \qquad n_z = 128, \qquad p \mapsto 4p$$

- Keep Courant number $\nu = c_g \Delta t / \Delta x \sim 10$ fixed†
  (i.e. Δt decreases)

$$\omega \propto \Delta t \propto \Delta x, \qquad \lambda^2 = \frac{1}{1 + (\alpha \Delta t)^2 (N^{*0})^2}$$

- All runs carried out on `Hector` Cray XE6 supercomputer
  2816 nodes of 2 $\times$ AMD Opteron 16-core Interlagos 2.3GHz = 90,122 cores

*OpenMP in vertical direction?

†NB explicit scheme requires $\nu \lesssim 1$

Weak Scaling

"Black box" AMG solvers: # iterations & time per iteration

Residual reduction: $||r||/||r_0|| \leq 10^{-5}$ all times in seconds

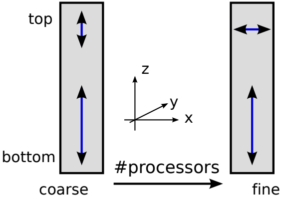
		AMG (DUNE)†			BoomerAMG†		
# proc	# dof	# iter	t_{iter}	eff.	# iter	t_{iter}	eff.
16	$8.3 \cdot 10^6$	11	0.63		12	0.73	
64	$3.4 \cdot 10^7$	11	0.64	[98%]	13	0.73	[100%]
256	$1.3 \cdot 10^8$	11	0.65	[97%]	12	0.75	[97%]
1024	$5.4 \cdot 10^8$	11	0.67	[94%]	12	0.75	[97%]
4096	$2.1 \cdot 10^9$	13	0.66	[95%]	12	0.75	[97%]
16384	$8.6 \cdot 10^9$	12	0.68	[92%]	11	0.86	[84%]
65536	$3.4 \cdot 10^{10}$	11	0.68	[92%]	9	2.24	[32%]

† as preconditioner for CG

Setup costs + Anisotropy

AMG has **coarse level** & **matrix** **setup costs**

Rotating anisotropy due to vertical grading



- Grid-aligned anisotropy
- Operator “well-behaved” in horizontal direction

⇒ **Tensor-product matrix-free geometric multigrid**

Börm, Hiptmair 2001. Numerical Algorithms. 26: 219234

Tensor-product multigrid

Tensor product operator

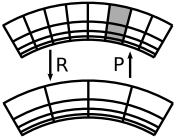
$$A \quad = \quad A^{(r)} \otimes M_h^{(horiz)} + M^{(r)} \otimes A_h^{(horiz)} \qquad \text{[for operator } \; -\nabla(\alpha\nabla\cdot) \text{]}$$

Vertical “eigenmodes”

$$A^{(r)} e_j^{(r)} = \omega_t M^{(r)} e_j^{(r)} \qquad u(r, \boldsymbol{x}) = \sum_{j=1}^{n_z} u_j(\boldsymbol{x}) e_j^{(r)}(r)$$

Börm, Hiptmair 2001. Numerical Algorithms. 26: 219234

- Vertical **line relaxation** (e.g. RB Gauss-Seidel)
- **Semi-coarsening** in horizontal direction only



$\Rightarrow$ 2d multigrid convergence rate

$$\rho^{(2d)} \quad \leftarrow \quad \max_j \left\{ \rho^{(horiz)} [e_j^{(r)}] \right\}$$

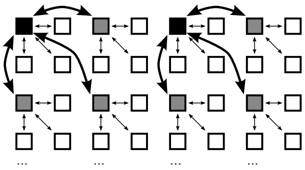
Meteorological application on 3d lat-lon grid:

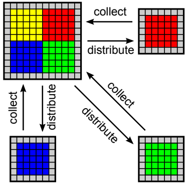
Buckeridge, Cullen, Scheichl and Wlasak 2011. Q J Royal Met Soc 137 (657):1083-1094.

Geometric multigrid

Implementation

- RB Line Gauss-Seidel ($1\times$ pre-/post-smoothing)
- Halo exchange after each smoothing step & prolongation
  $\Rightarrow$ Overlap calculation/communication
- collect/distribute coarse grid data when # procs > # columns





Geometric multigrid

Parallel Multigrid: volume/interface ratio decreases on coarser levels Hülsemann et al., Lect. Notes in Comp. Science and Engineering (2005)

BUT

Well conditioned on coarser levels ($-\omega^2 D^{(2)} u + u = RHS$)
Horizontal coupling vs. constant term:

$$4\frac{\omega^2}{\Delta x_\ell^2} \quad = \quad 4\frac{\omega^2}{\Delta x_0^2} \times 2^{-2\ell} \lesssim 2^{8-2\ell}$$

$\Rightarrow$ Reduce number of levels

- Coarsen to 1 column (standard MG)
- Coarsen to 1 column/processor (7 levels, *shallow* MG)
- 4 levels (*very shallow* MG)
- 1-level method to check robustness

Weak scaling results

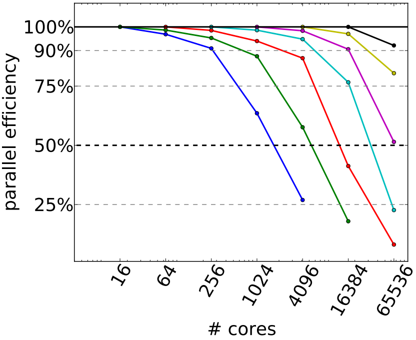
Different number of multigrid levels

all times in seconds

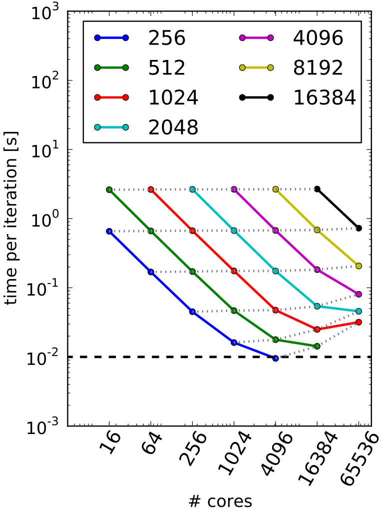
		standard MG		$n_{lev} = 7$		$n_{lev} = 4$	
# proc	# dof	#	t_{iter}	#	t_{iter}	#	t_{iter}
16	$8.3 \cdot 10^6$	6	0.332	6	0.332	6	0.333
64	$3.4 \cdot 10^7$	6	0.337 [99%]	6	0.335 [99%]	6	0.335 [99%]
256	$1.3 \cdot 10^8$	6	0.340 [98%]	6	0.338 [98%]	6	0.337 [99%]
1024	$5.4 \cdot 10^8$	6	0.343 [97%]	6	0.342 [97%]	5	0.340 [98%]
4096	$2.1 \cdot 10^9$	6	0.343 [98%]	6	0.340 [98%]	5	0.342 [97%]
16384	$8.6 \cdot 10^9$	6	0.350 [95%]	6	0.342 [97%]	5	0.342 [97%]
65536	$3.4 \cdot 10^{10}$	**6**	**0.373 [89%]**	**6**	**0.351 [95%]**	**5**	**0.342 [97%]**

Strong scaling results

Standard **geometric multigrid**
Problem size: $n \times n \times 256$

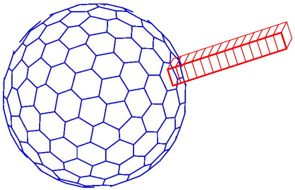


$$efficiency = \frac{p_0 \cdot T(p_0)}{p \cdot T(p)} \times 100\%$$



Multigrid on arbitrary spherical grids

Grid structure



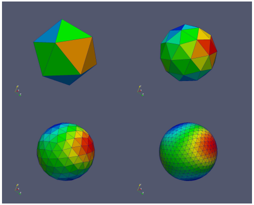
Tensor product grid structure

2-sphere ⊗ 1-column

host grid — directly addressed

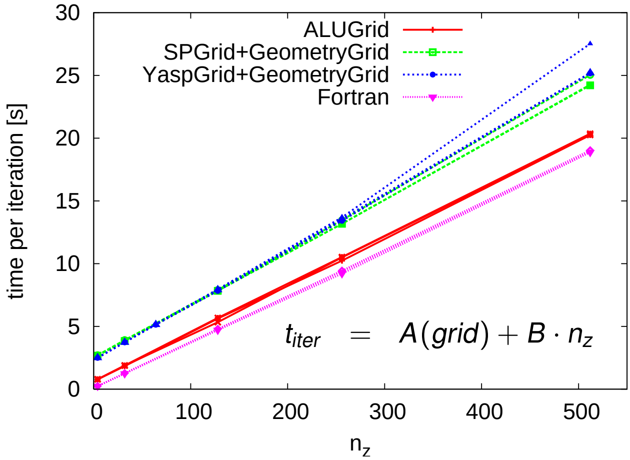
Size of vertical column $\mathcal{O}(100)$

- "Hide" indirect addressing in horizontal direction by work in vertical direction
  MacDonald et al., Int J of HPC Appl (2011)
- Naturally maps to DUNE data model: Attach vector of size n_z to each cell of the 2d host grid
- Multigrid hierarchy only on host grid



Comparison to DUNE geometric MG code

Time per iteration [Intel(R) Core(TM)2 Duo CPU E8400 3.00GHz]



Implemented together with Andreas Dedner (Warwick)

Summary and outlook

Summary

- Multigrid solvers for elliptic PDE in NWP implicit time stepping
- Verified **weak & strong scaling to 65536 cores (HECToR)**
  Access to bigger machines?
- Geometric multigrid code avoids AMG- and matrix setup costs
- **Anisotropy**: **Tensor product multigrid**
  semi-coarsening + vertical line relaxation
- Problem **well-conditioned on coarser grids**
  $\Rightarrow$ use small number of multigrid levels
- Geometric multigrid robust

Outlook

- Hybrid MPI+OpenMP parallelisation
- More realistic problems (ENDGame?):
  non-symmetry, non-smoothness,...
- GPGPUs