HIRLAM All Staff Meeting/ALADIN Workshop
1–5 April 2019, Madrid

A robust and scalable non-spectral solver for the ALADIN-NH dynamics

Daan Degrauwe

A scrambled presentation . . .

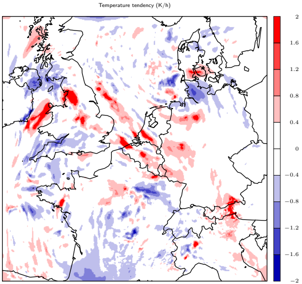
- Results

A scrambled presentation . . .

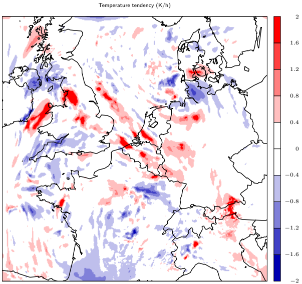
- Results
- The current spectral solver
- A non-spectral solver
- Some more results
- Conclusions and future work

- 500hPa Temperature tendency on a 6912×6912 grid with resolution 250 m and timestep of 7.5 s:

Current spectral solver

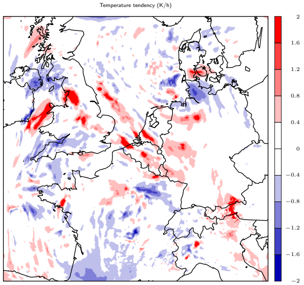


New non-spectral solver

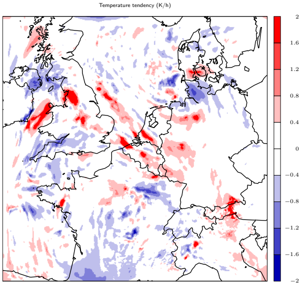


- 500hPa Temperature tendency on a 6912×6912 grid with resolution 250 m and timestep of 7.5 s:

Current spectral solver



New non-spectral solver



- On 46656 cores on ECMWF's Cray, the Helmholtz solver takes

0.6s/timestep **0.1s/timestep**

- We like our current dynamics a lot!
  - Accurate[1]
    due to spectral derivatives
  - Stable[2]
    due to semi-implicit semi-Lagrangian scheme
  - Efficient[3]
    due to large time steps
- For the implicit timestepping, a 3D Helmholtz problem needs to be solved:

$$\left(\mathbf{I} - \delta t^2 \nabla^2 \mathbf{B}_D^*\right) D^{t+\Delta t} = RHS$$

- This equation is first decoupled vertically, leading to $NLEV$ 2D Helmholtz problems:

$$\left(1 - c_\ell^2 \delta t^2 \nabla^2\right) \psi_\ell = rhs_\ell \qquad \text{for } \ell = 1, \ldots, NLEV$$

- These 2D Helmholtz problems are solved *in spectral space*

The problem at hand

- These 2D Helmholtz problems are solved *in spectral space*
- But spectral transforms require domain-wide information:

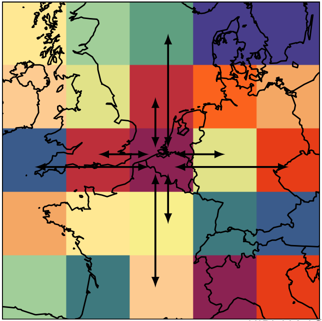
$$X_k = \sum_{j=0}^{N-1} x_j e^{2\pi i j k / N}$$

The problem at hand

- These 2D Helmholtz problems are solved *in spectral space*
- But spectral transforms require domain-wide information:

$$X_k = \sum_{j=0}^{N-1} x_j e^{2\pi i j k/N}$$

- On a distributed parallel computer, this means domain-wide communications:



- The need for domain-wide communications is a bit strange from a physical point of view: atmospheric waves don't travel that far in one time step.

- The need for domain-wide communications is a bit strange from a physical point of view: atmospheric waves don't travel that far in one time step.
- A conversation between a scientist and his hpc would look like this:

scientist: What are you doing ?

The problem at hand

- The need for domain-wide communications is a bit strange from a physical point of view: atmospheric waves don't travel that far in one time step.
- A conversation between a scientist and his hpc would look like this:

scientist: What are you doing ?

computer: I'm computing.

The problem at hand

- The need for domain-wide communications is a bit strange from a physical point of view: atmospheric waves don't travel that far in one time step.
- A conversation between a scientist and his hpc would look like this:

scientist: What are you doing ?

computer: I'm computing.

Why is it taking so long?

The problem at hand

- The need for domain-wide communications is a bit strange from a physical point of view: atmospheric waves don't travel that far in one time step.
- A conversation between a scientist and his hpc would look like this:

scientist: What are you doing ?

computer: I'm computing.

Why is it taking so long?

Just making sure the results are accurate. . .

The problem at hand

- The need for domain-wide communications is a bit strange from a physical point of view: atmospheric waves don't travel that far in one time step.
- A conversation between a scientist and his hpc would look like this:

scientist: What are you doing ?

computer: I'm computing.

Why is it taking so long?

Just making sure the results are accurate...

Still not finished !?

The problem at hand

- The need for domain-wide communications is a bit strange from a physical point of view: atmospheric waves don't travel that far in one time step.
- A conversation between a scientist and his hpc would look like this:

scientist: What are you doing ?

computer: I'm computing.

Why is it taking so long?

Just making sure the results are accurate. . .

Still not finished !?

Nope, working on the 40th digit of the temperature forecast. . .

The problem at hand

- The need for domain-wide communications is a bit strange from a physical point of view: atmospheric waves don't travel that far in one time step.
- A conversation between a scientist and his hpc would look like this:

scientist: What are you doing ?

computer: I'm computing.

Why is it taking so long?

Just making sure the results are accurate. . .

Still not finished !?

Nope, working on the 40th digit of the temperature forecast. . .

. . . ?

The problem at hand

- The need for domain-wide communications is a bit strange from a physical point of view: atmospheric waves don't travel that far in one time step.
- A conversation between a scientist and his hpc would look like this:

  scientist: What are you doing ?

  computer: I'm computing.

  Why is it taking so long?

  Just making sure the results are accurate. . .

  Still not finished !?

  Nope, working on the 40th digit of the temperature forecast. . .

  . . . ?

- In the meantime, we are debating whether we single (or even lower) precision isn't sufficient. . .

A non-spectral solver

- When formulated in gridpoint space, the 2D Helmholtz problems become large sparse linear systems.
- These can be solved by *preconditioned iterative solvers*

A non-spectral solver

- When formulated in gridpoint space, the 2D Helmholtz problems become large sparse linear systems.
- These can be solved by *preconditioned iterative solvers*
- Key questions:
  1. What kind of solver/preconditioner to use?
  2. What about impact on the rest of the model?
  3. What about robustness?
  4. What about efficiency?
  5. What about accuracy?

A non-spectral solver: solver type

- A small test program (dwarf) was developed to compare several solvers for the Helmholtz problem:
  - Spectral solver
  - GCR(k) Krylov solver
  - Richardson solver

    (somewhat slower than Krylov solvers, but it does not need scalar products)

- Preconditioners that were implemented are:
  - Gauss-Seidel preconditioner

    Efficient to reduce the small-scale errors
  - Multigrid preconditioner

    Efficient to reduce the large-scale errors

    Contains several parameters: depth, number of pre-, bottom- and post-relaxations

A non-spectral solver: model impact

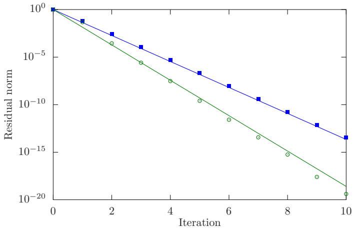
- The equations, the spectral transforms and the distribution mechanisms in the test program are exactly the ones from ALADIN-HIRLAM
- Therefore, the iterative solvers should be compatible with the rest of the model:
  - A-grid horizontal discretization
  - mass-based coordinate
  - semi-Lagrangian advection
  - physics
  - ...

A non-spectral solver: robustness

- For our constant-coefficient 2D Helmholtz problems, the convergence speed of a given iterative solver only depends on the wave Courant number

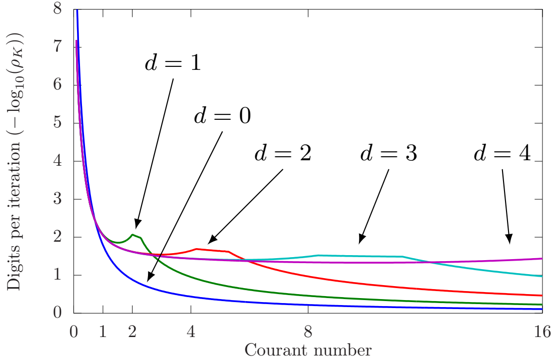
$$\mu_\ell = \frac{c_\ell \Delta t}{\Delta x}$$

- Convergence speed does not depend on domain size or weather conditions!
- Therefore, the number of iterations required to reach a given precision is known beforehand!



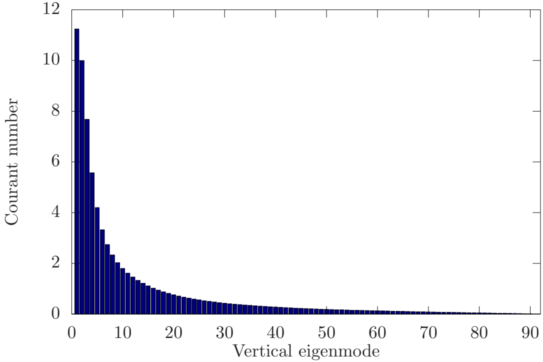
A non-spectral solver: efficiency

- The same property (predictable convergence speed) can be used to choose optimal multigrid preconditioner parameters:



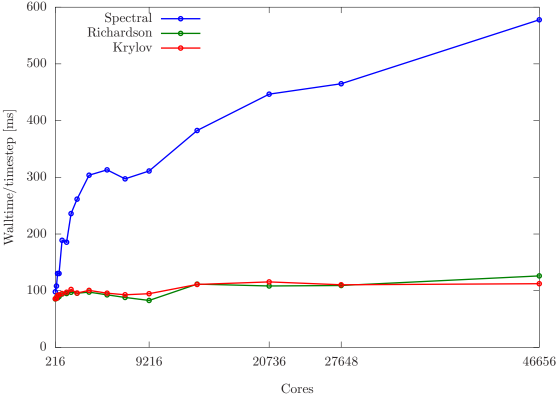
Number of digits gained per iteration with a Krylov solver as a function of the Courant number, for various multigrid depths d.

- Different optimal preconditioner parameters can be chosen for the different vertical modes (each of which has its own Courant number)



- Nothing beats the accuracy of spectral derivatives.
- Previous work has shown that higher-order finite differences on an A-grid also perform well.
- Accuracy is definitely a concern and should be investigated further!

- Weak scalability tests on ECMWF's Cray:



Important note: only scalability of Helmholtz solver!

Conclusions and future work

- The accuracy of a spectral model comes at a price:
  - Is such accuracy necessary?
  - Scalability of the spectral transforms is problematic, albeit not in the immediate future for our domains.
- As it turns out, specific properties of our NH dynamics can be used to greatly improve the performance of iterative solvers:
  - constant-coefficient semi-implicit

    $\Rightarrow$ predictable convergence = **robustness**
  - vertical decoupling

    $\Rightarrow$ optimal preconditioner parameters = **efficiency**
- The scalability of the preconditioned iterative solvers is really good!

Conclusions and future work

- A lot of work remains to be done:
  - Integrate in ALADIN-HIRLAM code
  - Other usage of spectral derivatives, e.g. diffusion
  - Accuracy impact!
  - Implicit treatment of orography?
  - Usage of Atlas?

Thank you