Global, non-hydrostatic, cloud-permitting medium-range forecasts: progress and challenges

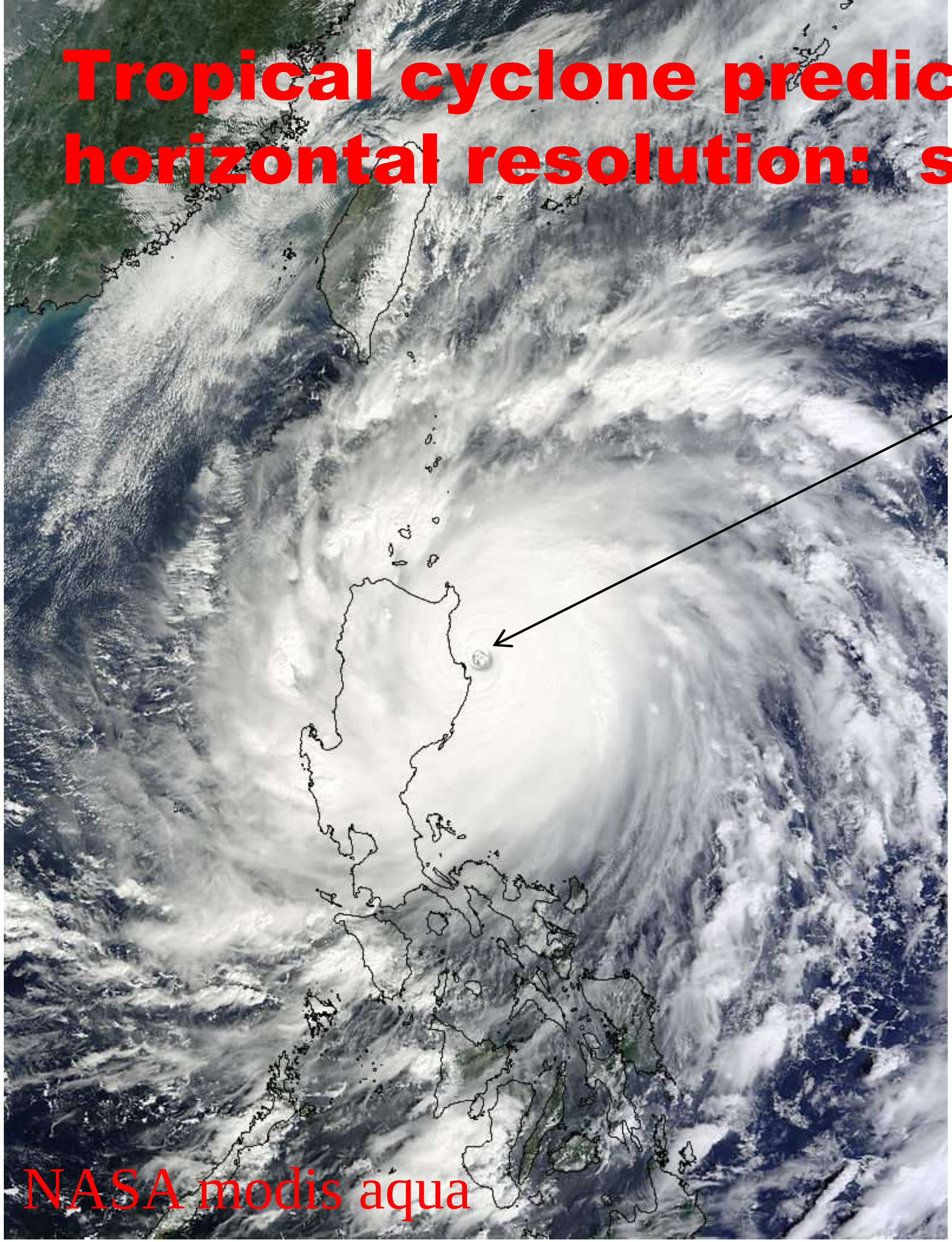
Nils Wedi, Mats Hamrud, George Mozdzynski

Thanks to Jim Doyle for bringing Megi to my attention

Introduction – A history

- Resolution increases of the deterministic 10-day medium-range Integrated Forecast System (IFS) over ~28 years at ECMWF:
  - 1983: T 63 (~316km)
  - 1987: T 106 (~188km)
  - 1991: T 213 (~95km)
  - 1998: T_L319 (~63km)
  - 2000: T_L511 (~39km)
  - 2006: T_L799 (~25km)
  - 2010: T_L1279 (~16km)
  - 2015?: T_L2047 (~10km)
  - 2020-???: (~1-10km) Non-hydrostatic, cloud-permitting, substantially different cloud-microphysics and turbulence parametrization, substantially different dynamics-physics interaction ?

Tropical cyclone prediction and high horizontal resolution: super typhoon Megi



NASA modis aqua

Observed core pressure:

885hPa *Wind speeds ~ 70-80m/s*

(one of the lowest ever recorded)

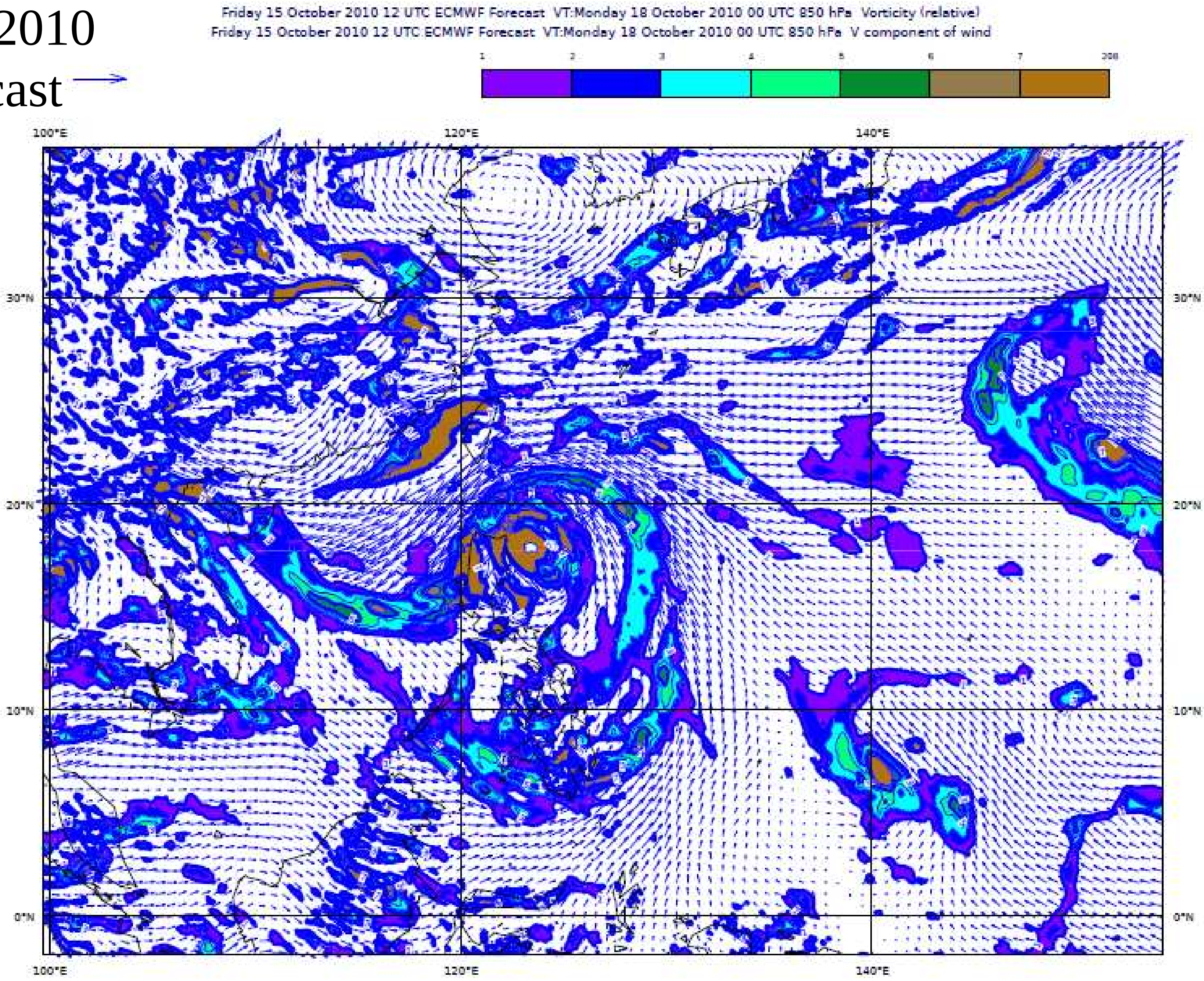
Model core pressure (T1279):

~955hPa *Wind speeds ~ 60m/s*

18th October 2010

2.5 day forecast

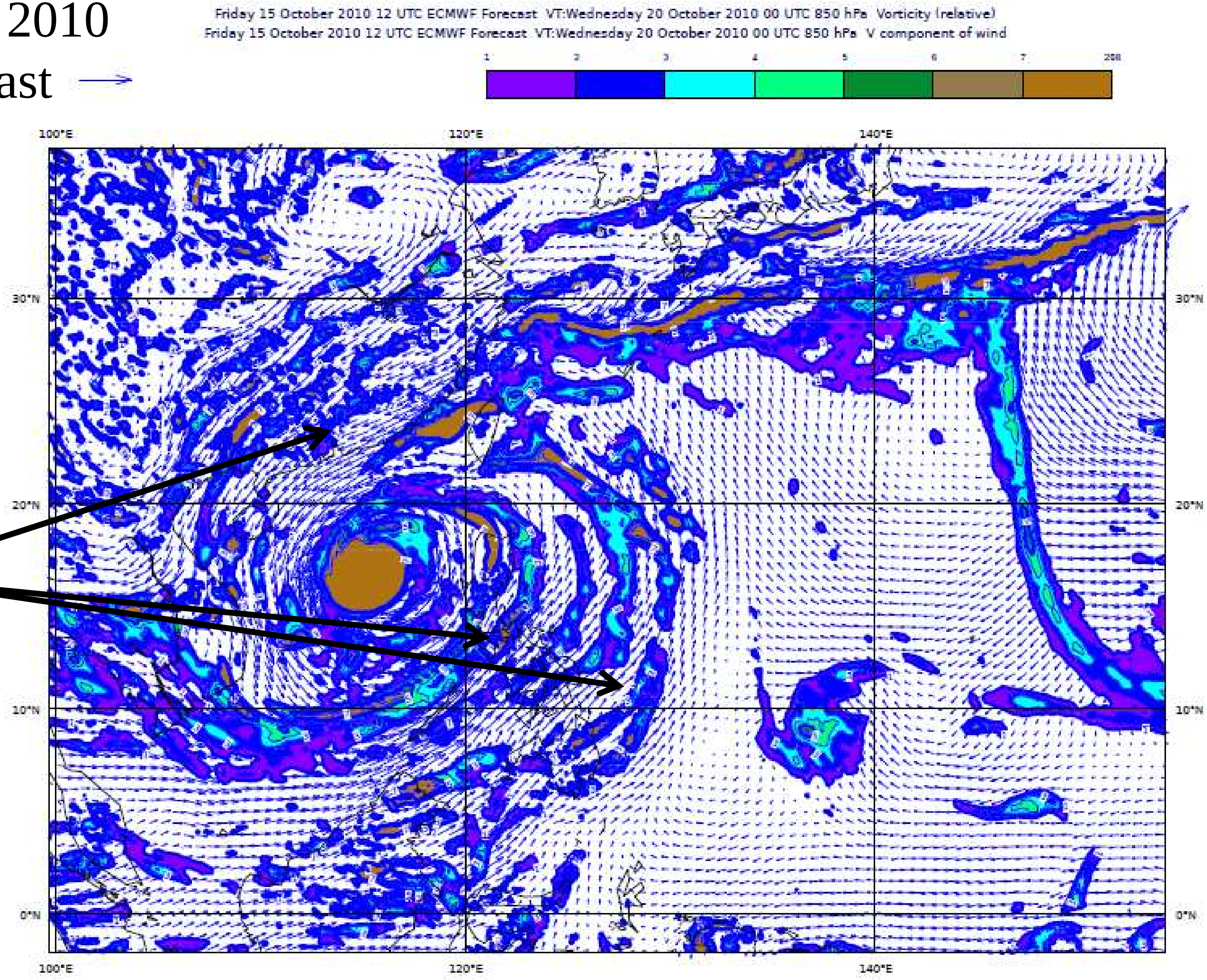
Core pressure
Error ~70hPa


Friday 15 October 2010 12 UTC ECMWF Forecast VT:Monday 18 October 2010 00 UTC 850 hPa Vorticity (relative)
Friday 15 October 2010 12 UTC ECMWF Forecast VT:Monday 18 October 2010 00 UTC 850 hPa V component of wind

20th October 2010
4.5 day forecast →

Core pressure error reduces to 10-20hPa

concentric vorticity bands



Typhoon Megi simulations with IFS

- *IFS shows a good medium-range cyclone track forecast of Megi and reveals interesting vortex dynamics, but large error in core pressure !*
- *Hydrostatic as well as non-hydrostatic simulations at T2047 (~10km) show a similar core pressure error !*
- *Intercomparison of NICAM and IFS in the Athena project suggests potential advantages (and disadvantages) of explicitly simulated (even if not necessarily resolved) vs. parametrized convection*

The Athena Project (6 months)

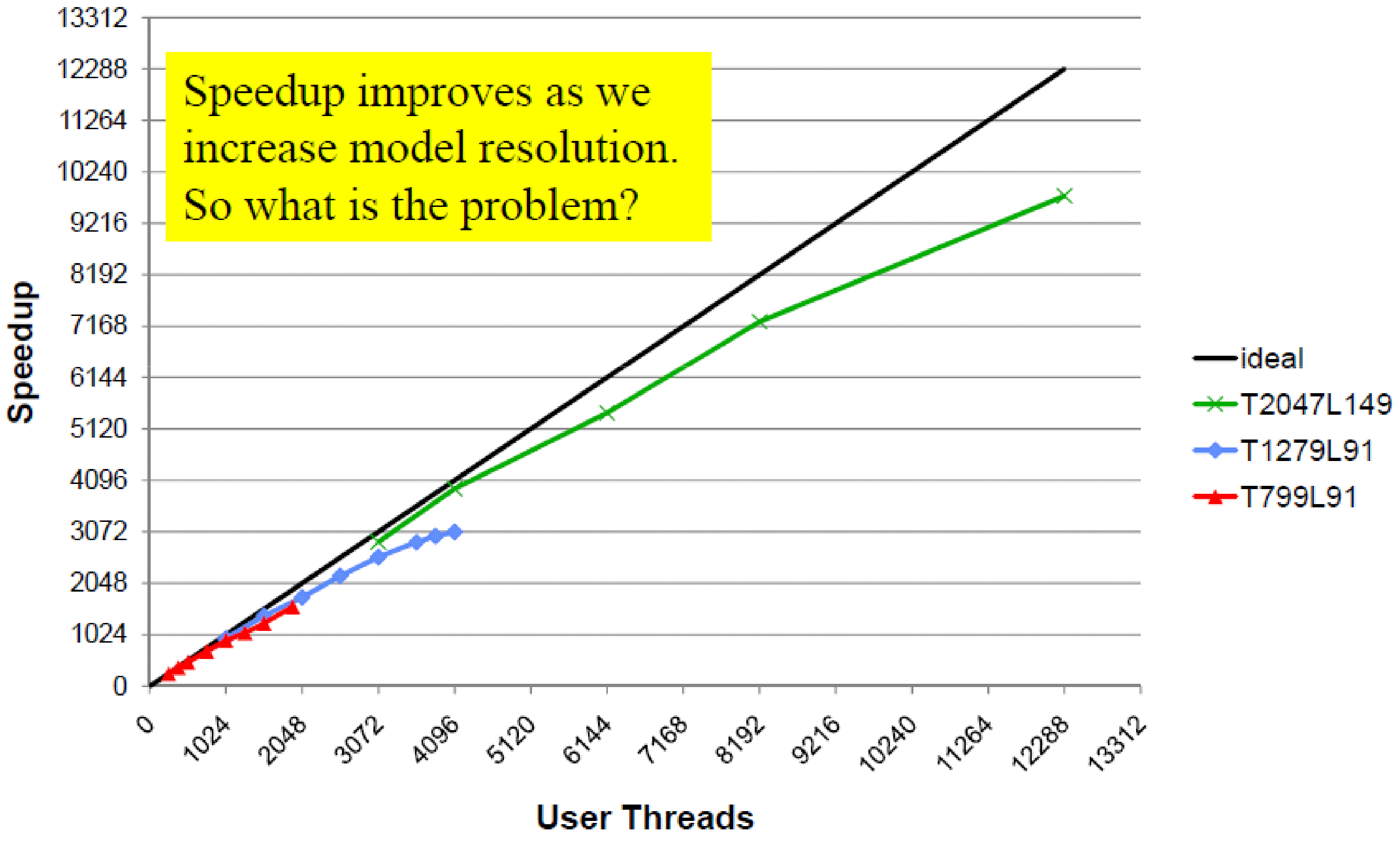
(not only) an example of the computational efficiency of the hydrostatic IFS

- IFS (cycle 36r1) atmosphere-only runs with prescribed SST data from observations until 2007 (2070- A1B scenario SST forcing comes from CCSM simulation)

- Set of 13-months long integrations (1960-2007) and AMIP long runs (1960-2007 and 2070-2117)
    - T_L159L91 (~125 km, Δt = 3600s) 3 x 47 years
    - T_L511L91 (~39 km, Δt = 900s) 1 x 47 years
    - T_L1279L91 (~16 km, Δt = 600s) 3 x 47 years
    - T_L2047L91 (~10 km, Δt = 450s) 1x 19 years

- Factor 10-15 larger time-step compared to existing state-of-the-art non-hydrostatic models at equivalent resolutions, additional computational savings from the reduced grid (~30%) and the direct solver in the semi-implicit scheme.

IFS speedup on Power6 (+ T2047L149 RAPS11)



Speedup improves as we increase model resolution.
So what is the problem?

- -Increased resolution requires smaller time-step (bad for scaling)
- -Trend towards massively increased # of processors per core
- -Cost of communications
- -Data assimilation

The spectral transform method

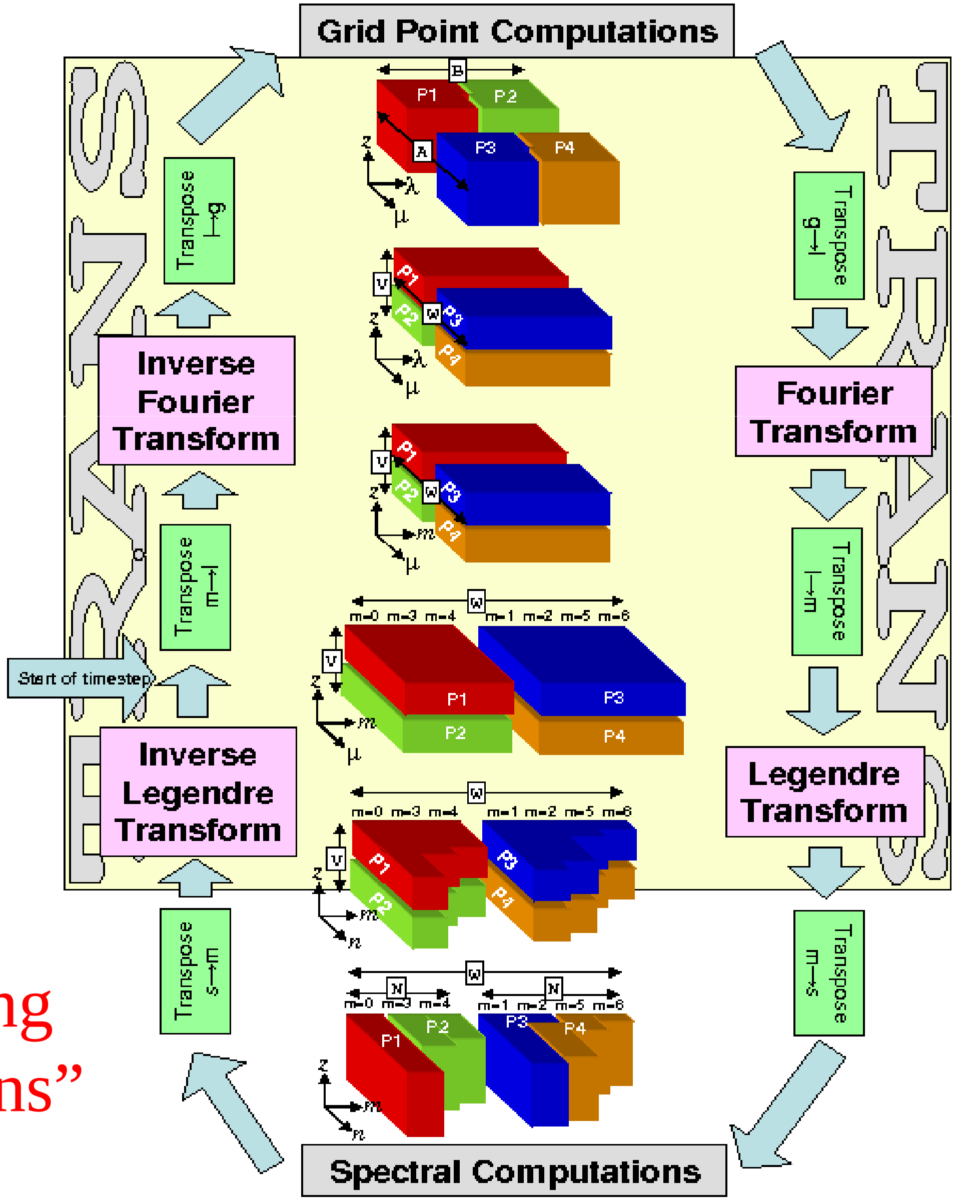
Eliasen et. al (1970), Orszaag (1970)

Applied at ECMWF for the last 30 years …

Spectral semi-Lagrangian semi-implicit (compressible) a viable option ?

-Computational efficiency on future MPP architectures ?
-Accuracy at cloud-resolving scales ?
-Suitable for the likely mixture of medium and high resolution ensembles and ultra-high resolution forecasts ?

Transpositions within the spectral transforms



Grid Point Computations

Transpose g→l

Fourier Transform

Transpose l→m

Legendre Transform

Transpose m→s

Spectral Computations

Transpose s→m

Inverse Legendre Transform

Transpose m→l

Inverse Fourier Transform

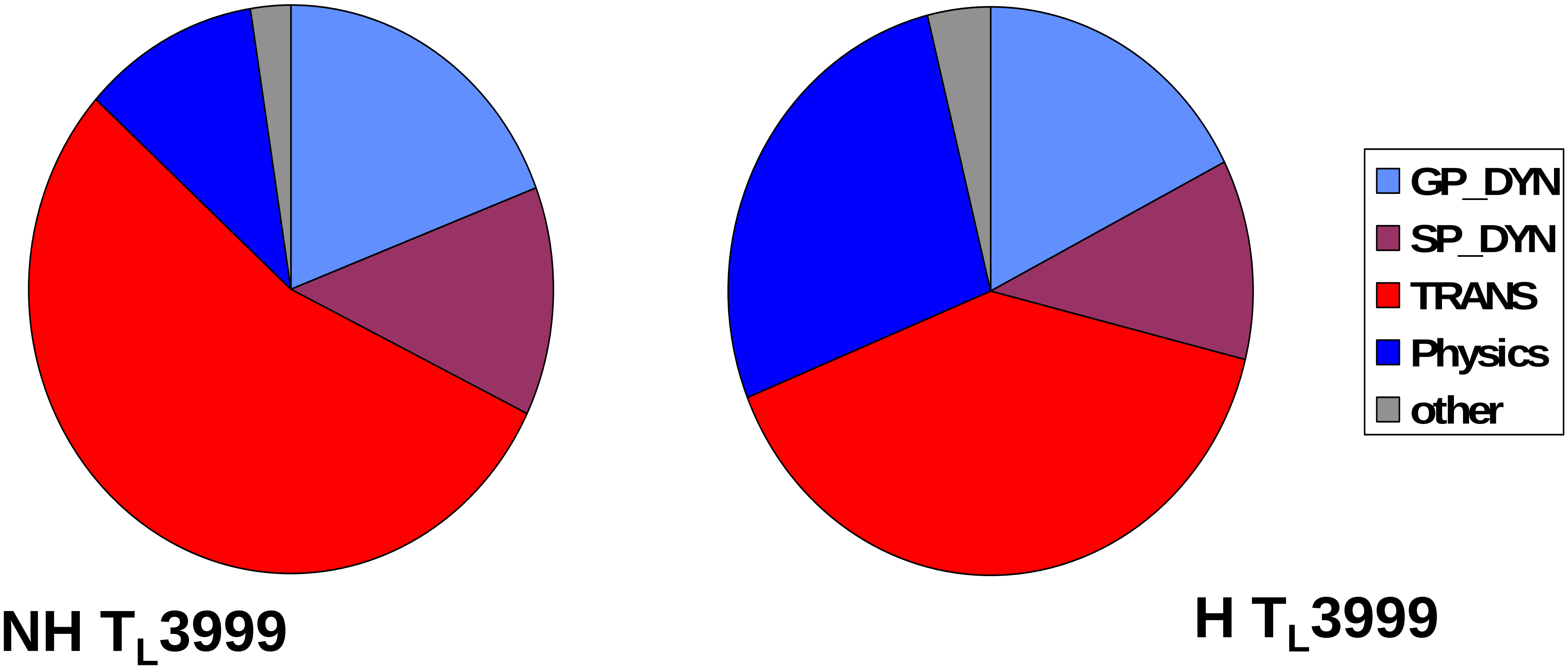
Transpose l→g

Start of timestep

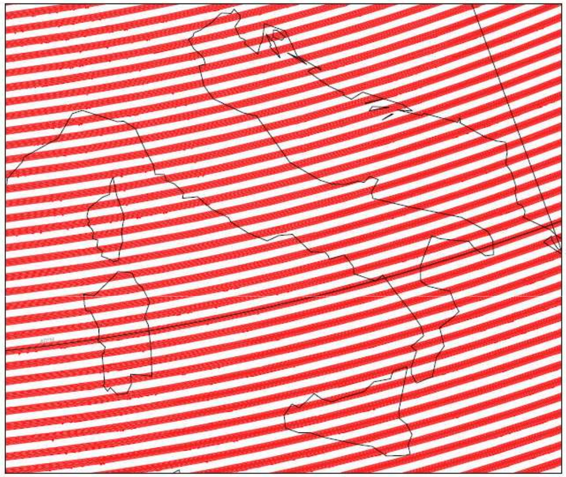
The time spent in message passing associated with the “transpositions” at T2047 is roughly equal to the computational time.

ECMWF

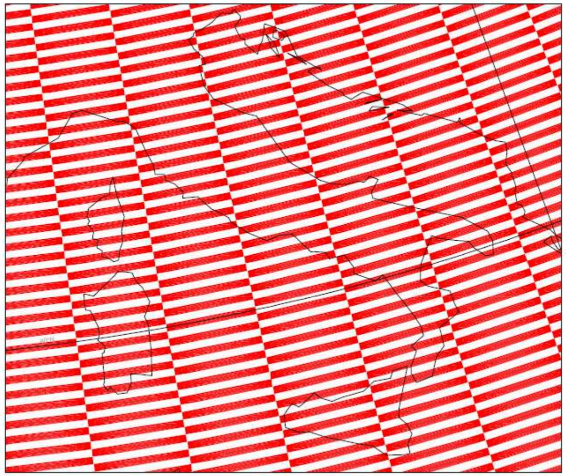
Computational Cost at T_L3999 hydrostatic vs. non-hydrostatic IFS



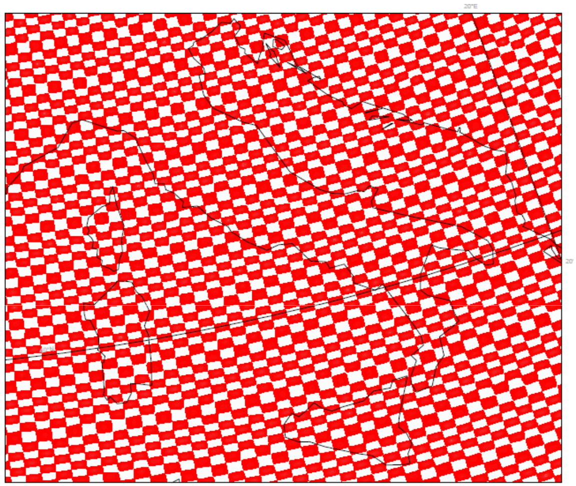
Spherical harmonics



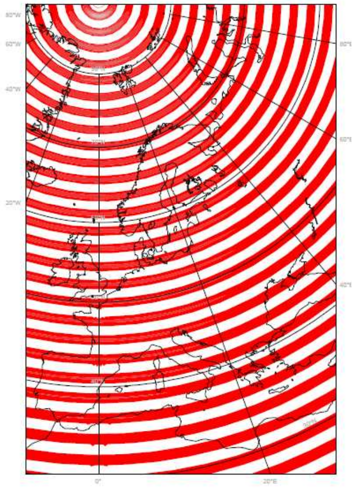
$Y_{n=1280}^{m=1}(\lambda,\mu)$



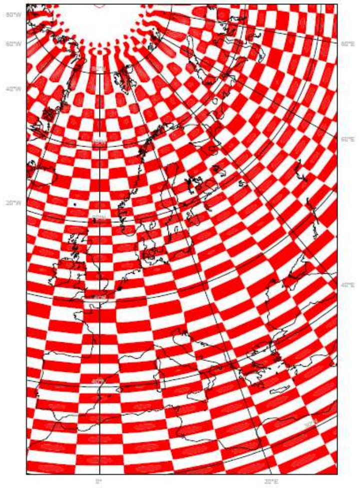
$Y_{n=1280}^{m=30}(\lambda,\mu)$



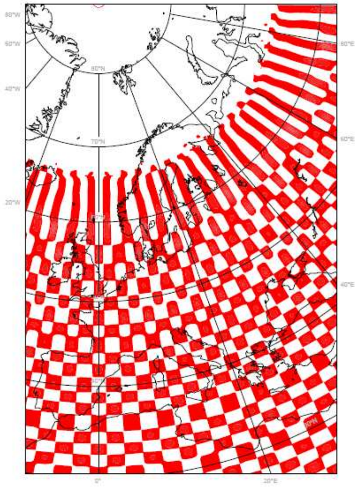
$Y_{n=1280}^{m=80}(\lambda,\mu)$



$Y_{n=160}^{m=1}(\lambda,\mu)$

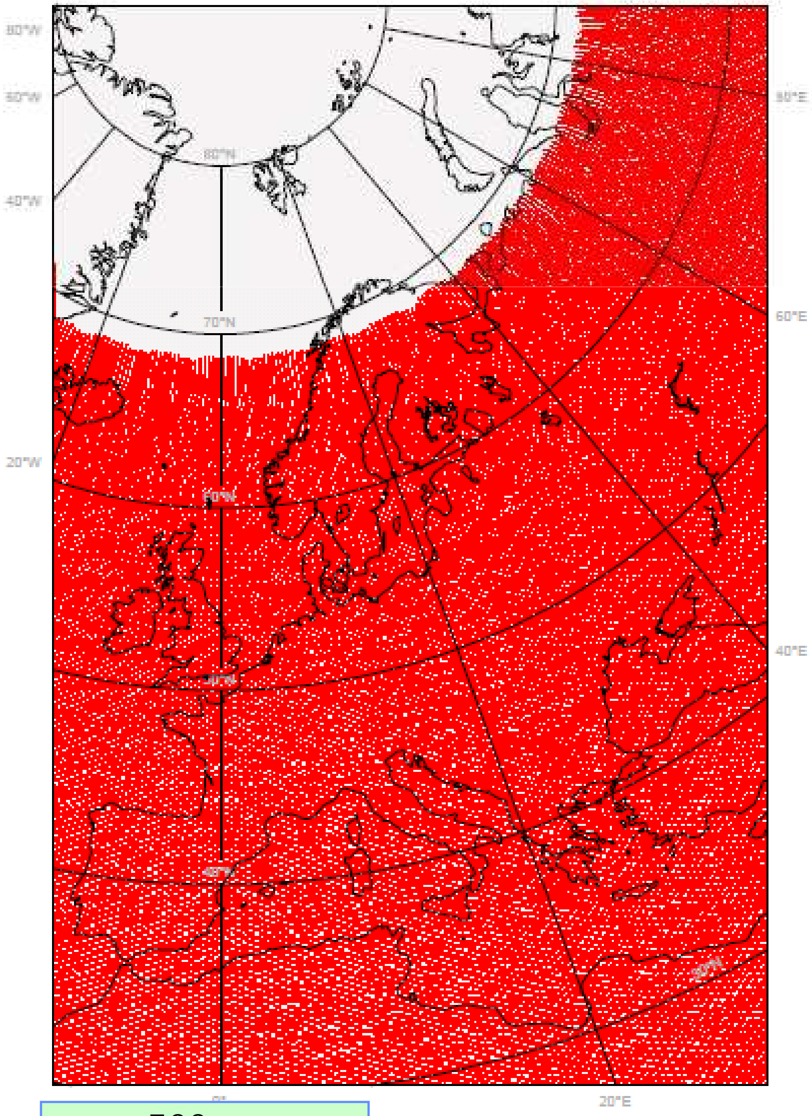


$Y_{n=160}^{m=30}(\lambda,\mu)$

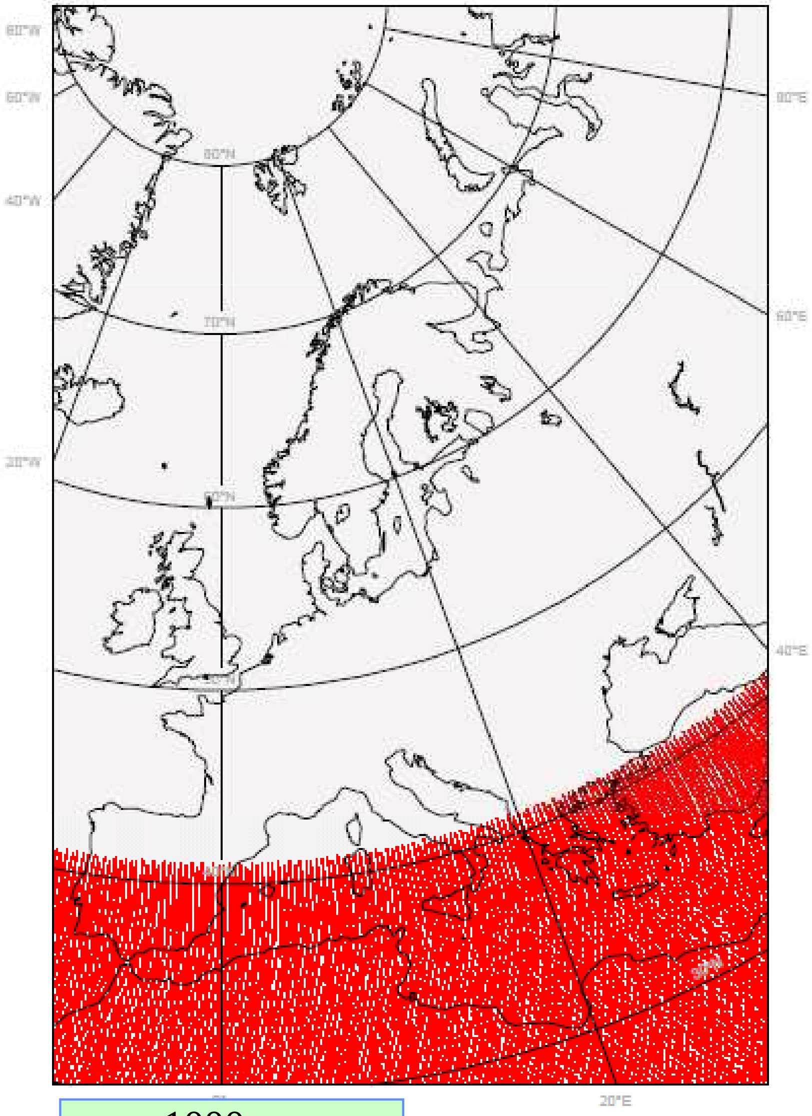


$Y_{n=160}^{m=80}(\lambda,\mu)$

Smallness of associated Legendre polynomials towards the poles at T1279



$Y_{n=1280}^{m=500}(\lambda,\mu)$



$Y_{n=1280}^{m=1000}(\lambda,\mu)$

Fast Legendre transform

◆ **The algorithm proposed in *(Tygert, 2008)* suitably fits into the IFS transform library by simply replacing the single DGEMM call with 2 new steps plus more expensive pre-computations.**

◆ **(1) Instead of the recursive *Cuppen divide-and-conquer algorithm (Tygert, 2008)* we use the so called *butterfly algorithm (Tygert, 2010)* based on a matrix compression technique via rank reduction with a specified accuracy to accelerate the arising *matrix-vector multiplies (sub-problems still use dgemm).***

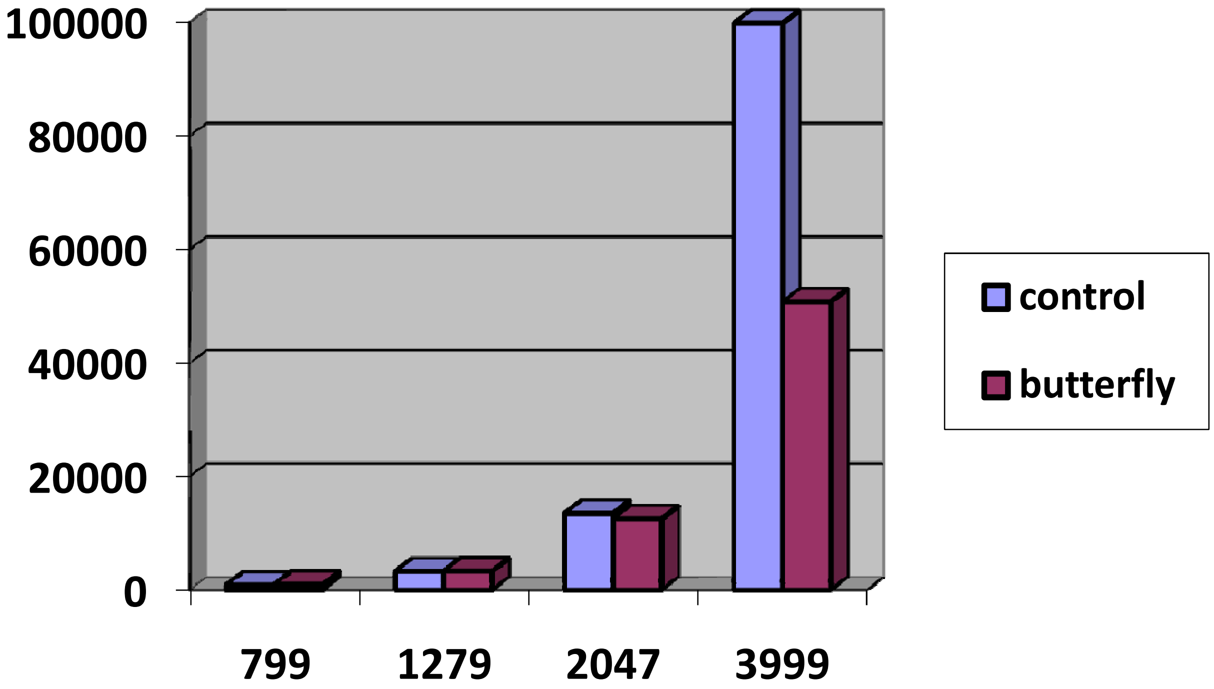
◆ **(2) The arising interpolation from one set of roots of the associated Legendre polynomials to another can be accelerated by using a *FMM (fast multipole method).***

Pre-computations

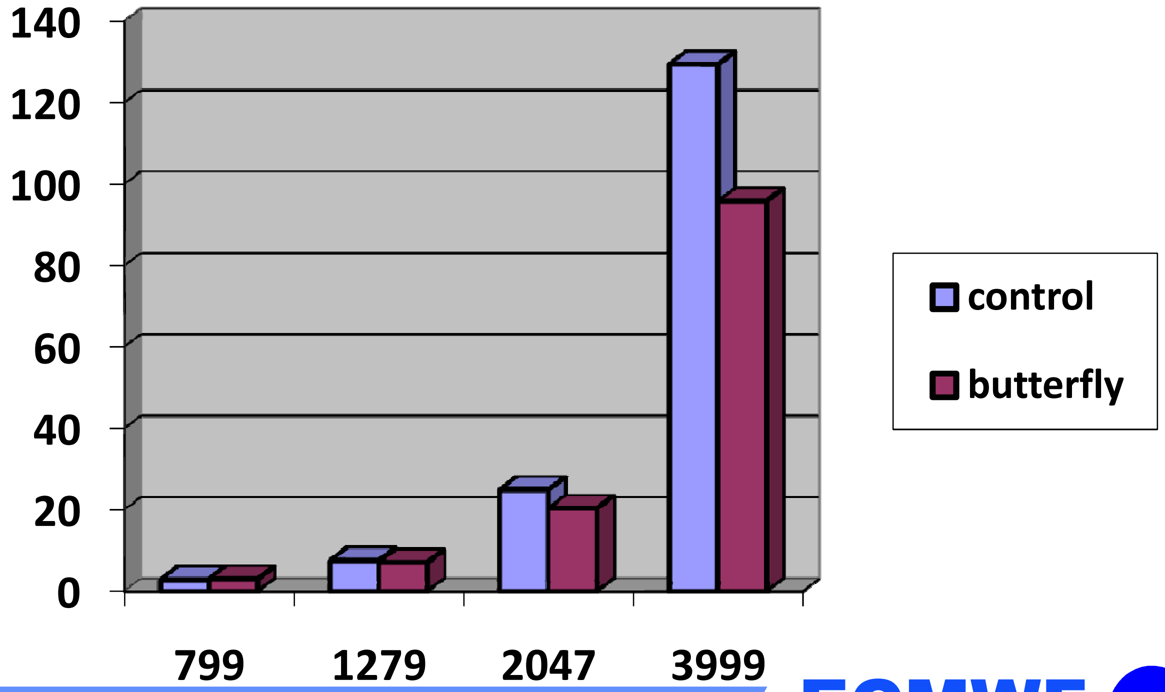
- **Large number of pre-computations necessary but the actual storage in memory required during the time-stepping is low!**
- **Accurate associated pre-computations can now be done in <1min at T3999 and these computations scale with the number of processors.**
  - **This major improvement in cost and accuracy was helped by using a recursive formula for the Legendre polynomials for a single *m* *(unlike Belousov, 1962).* By tracking the size of the exponentials and accurate direct starting formulas found in an “Antiquariat”!**

(Egersdoerfer, Reichsamt fuer Wetterdienst, Berlin 1936)

Floating point operations per time-step in Gflop

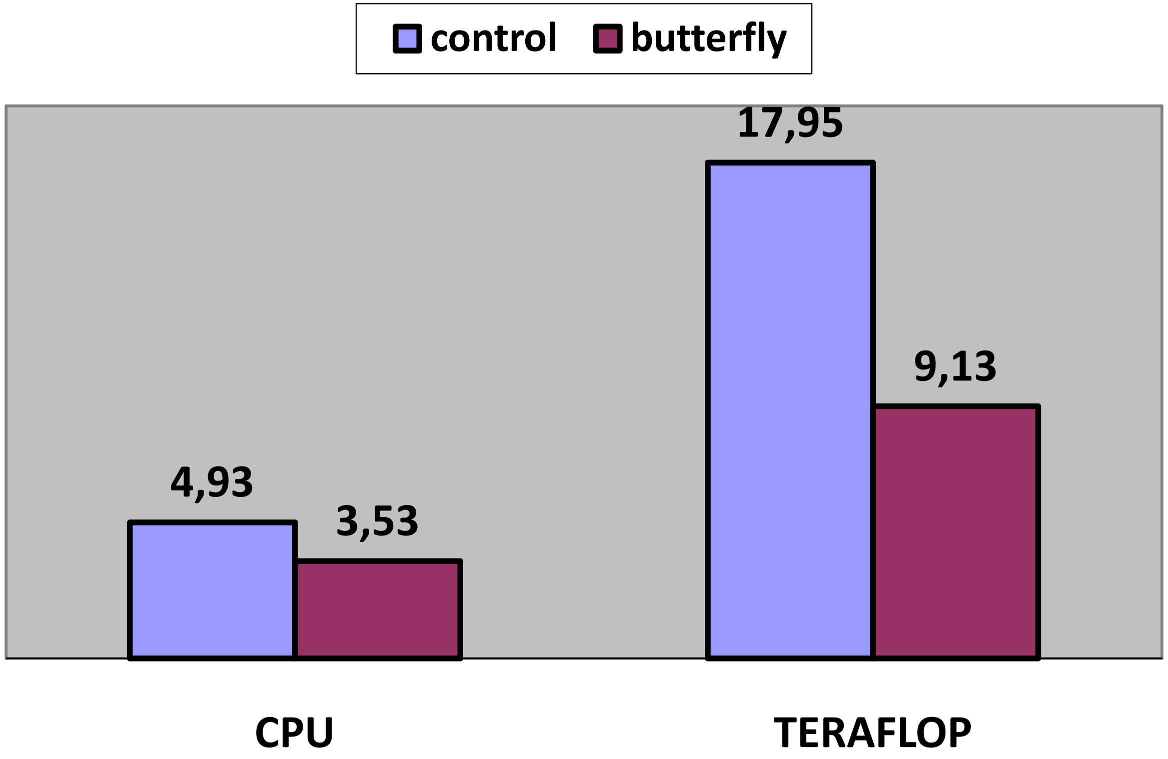


Computational cost per time-step in seconds

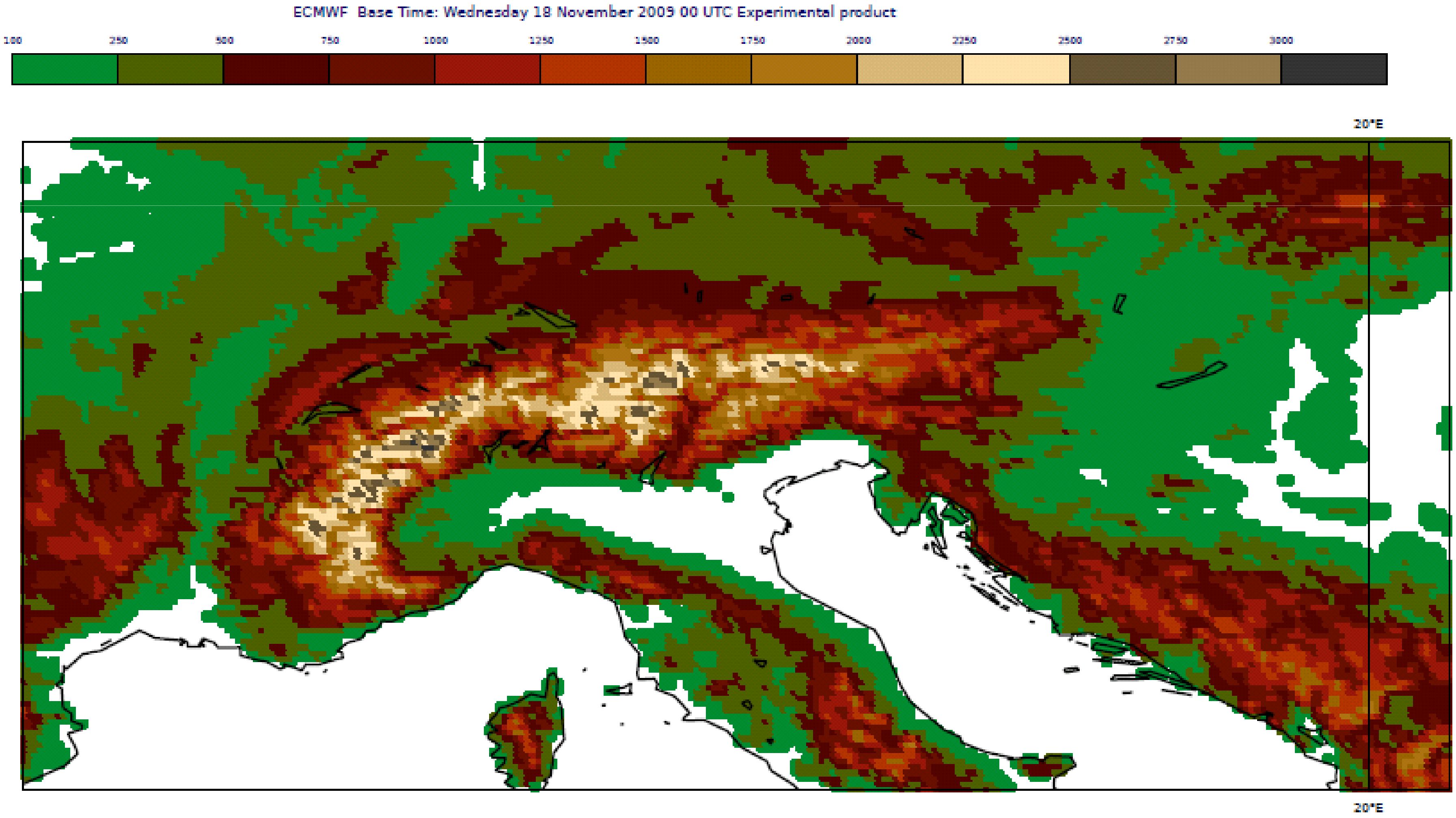


Inverse transform of single field

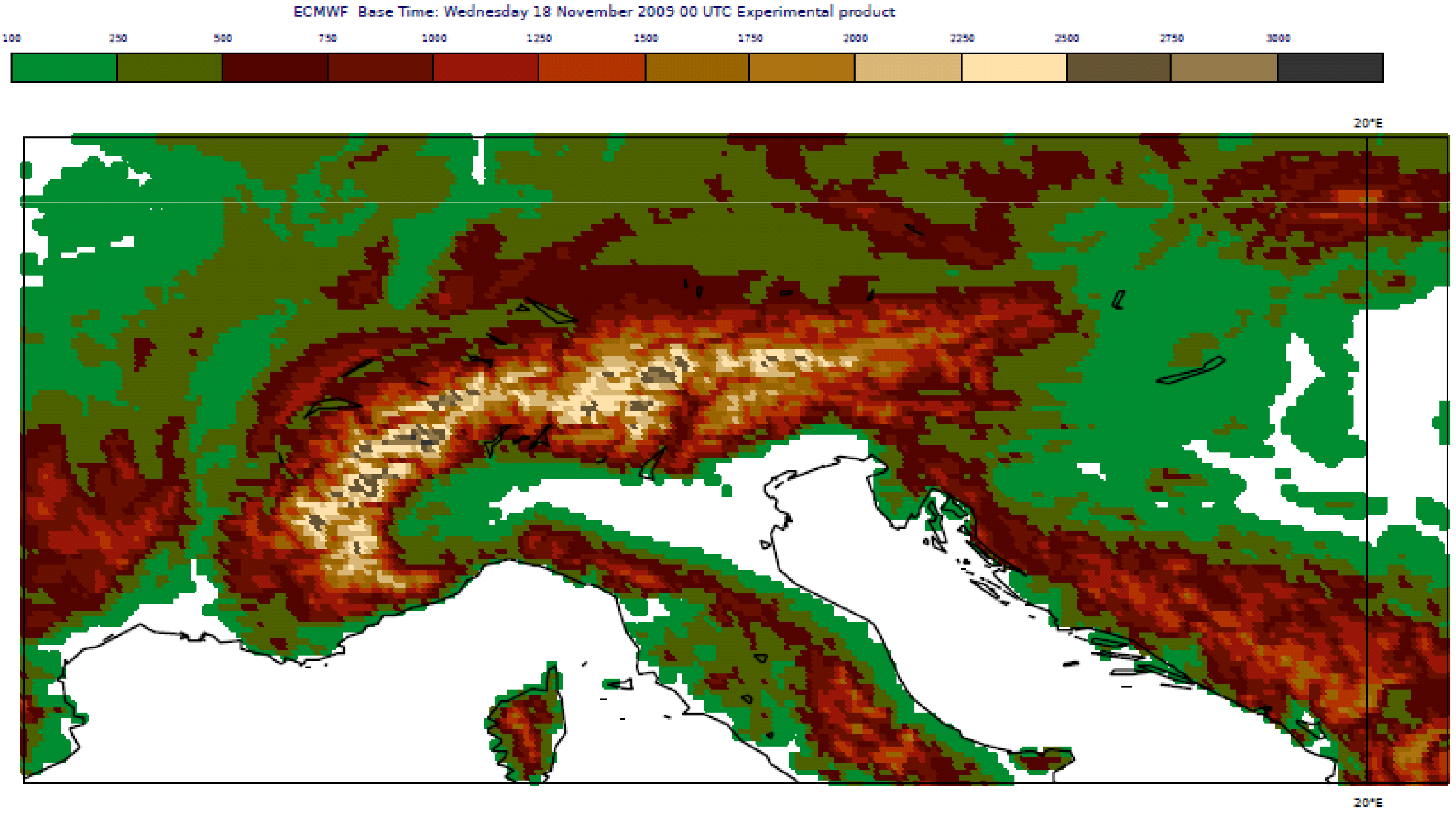
T3999 6h forecast – inverse transform



T3999 control inverse transform



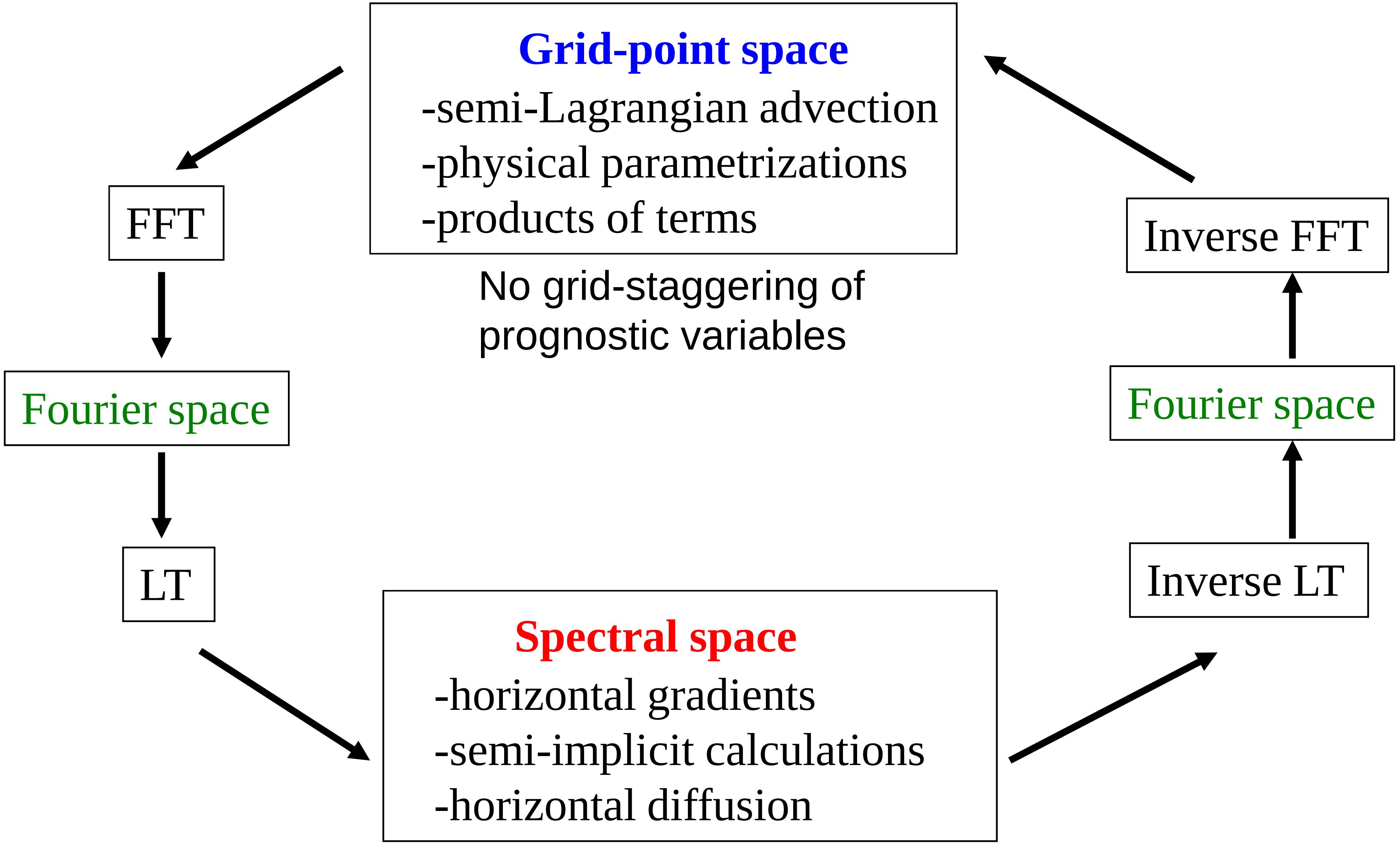
T3999 butterfly inverse transform



Summary

- Need to be ready for global cloud-resolving simulations well before an operational implementation for research development, still much room for improvement …
- IFS still computationally very efficient at T2047 !
- Fast Legendre Transform *(Tygert, 2008,2010)* for the first time shown to work truly faster in IFS beyond T2047 !

Schematic description of the spectral transform method in the ECMWF IFS model



Grid-point space

- -semi-Lagrangian advection
- -physical parametrizations
- -products of terms

No grid-staggering of prognostic variables

FFT

Fourier space

LT

Spectral space

- -horizontal gradients
- -semi-implicit calculations
- -horizontal diffusion

Inverse LT

Fourier space

Inverse FFT

FFT: Fast Fourier Transform, LT: Legendre Transform

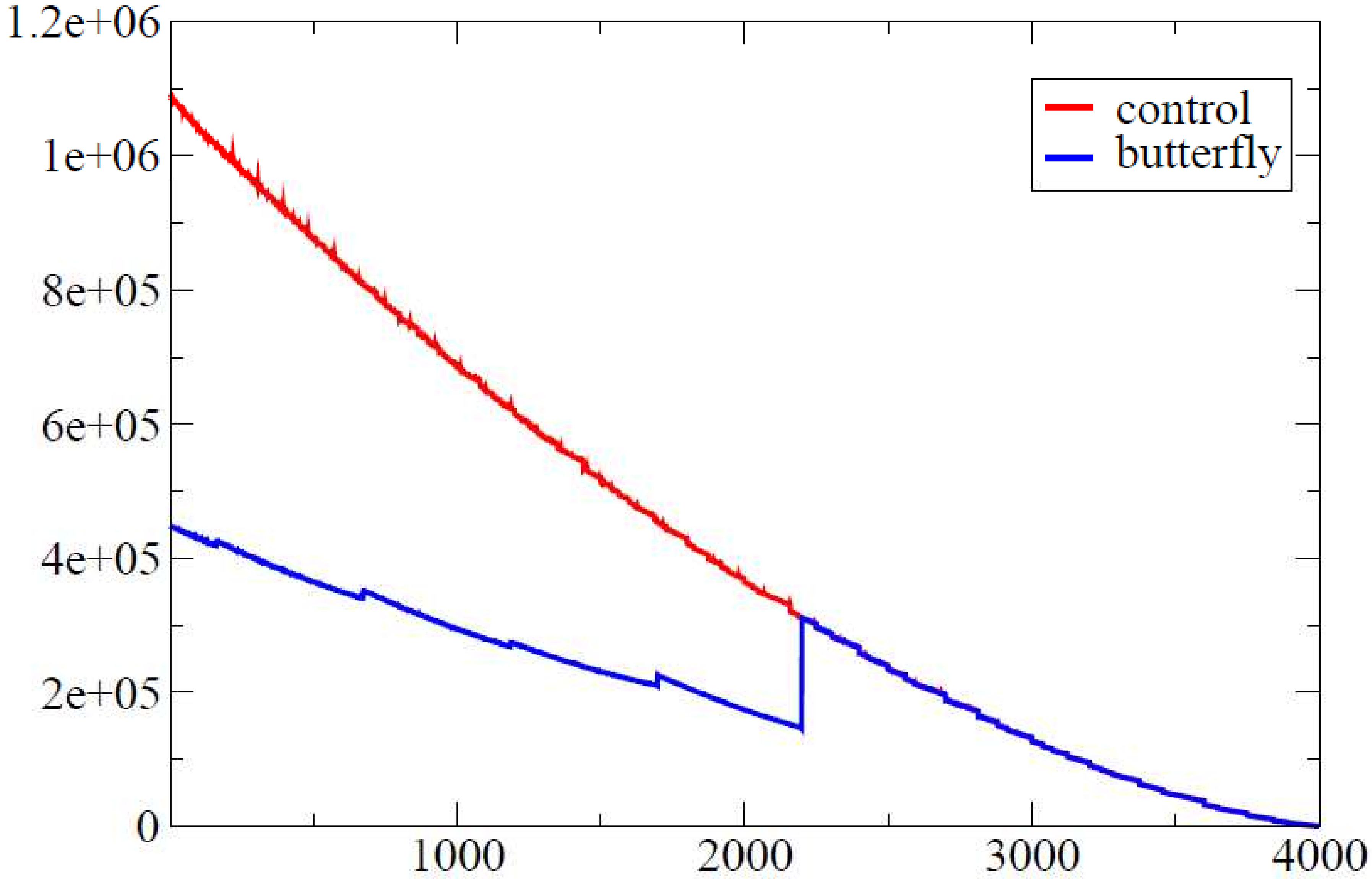
Cost of the spectral transform method

- **FFT can be computed as C*N*log(N) where C is a small positive number and N is the cut-off wave number in the triangular truncation.**
- **Ordinary Legendre transform is $O(N^2)$ but can be combined with the fields/levels such that the arising matrix-matrix multiplies make use of the highly optimized BLAS routine DGEMM.**
- **But overall cost is $O(N^3)$ for both memory and CPU time requirements.**

Desire to use a fast Legendre transform where the cost is proportional to C*N*log(N) with C << N

and thus overall cost N^2*log(N)

T3999 6h forecast - inverse transform: Floating point operations vs. wave number



T3999 6h forecast - inverse transform: CPU time vs. wave number