Inter-Processor communication patterns in weather forecasting models

Tomas Wilhelmsson

Swedish Meteorological and Hydrological Institute

Sixth Annual Workshop on

Linux Clusters for Super Computing

2005-11-18

SMHI

Numerical Weather Prediction

Analysis

Obtain best estimate of current weather situation from

2. Background, (the last forecast 6 to 12 hours ago)
3. Observations (ground, aircraft, ship, radiosondes, satellites)

Variational assimilation in 3D or 4D

Most computationally expensive part

Forecast

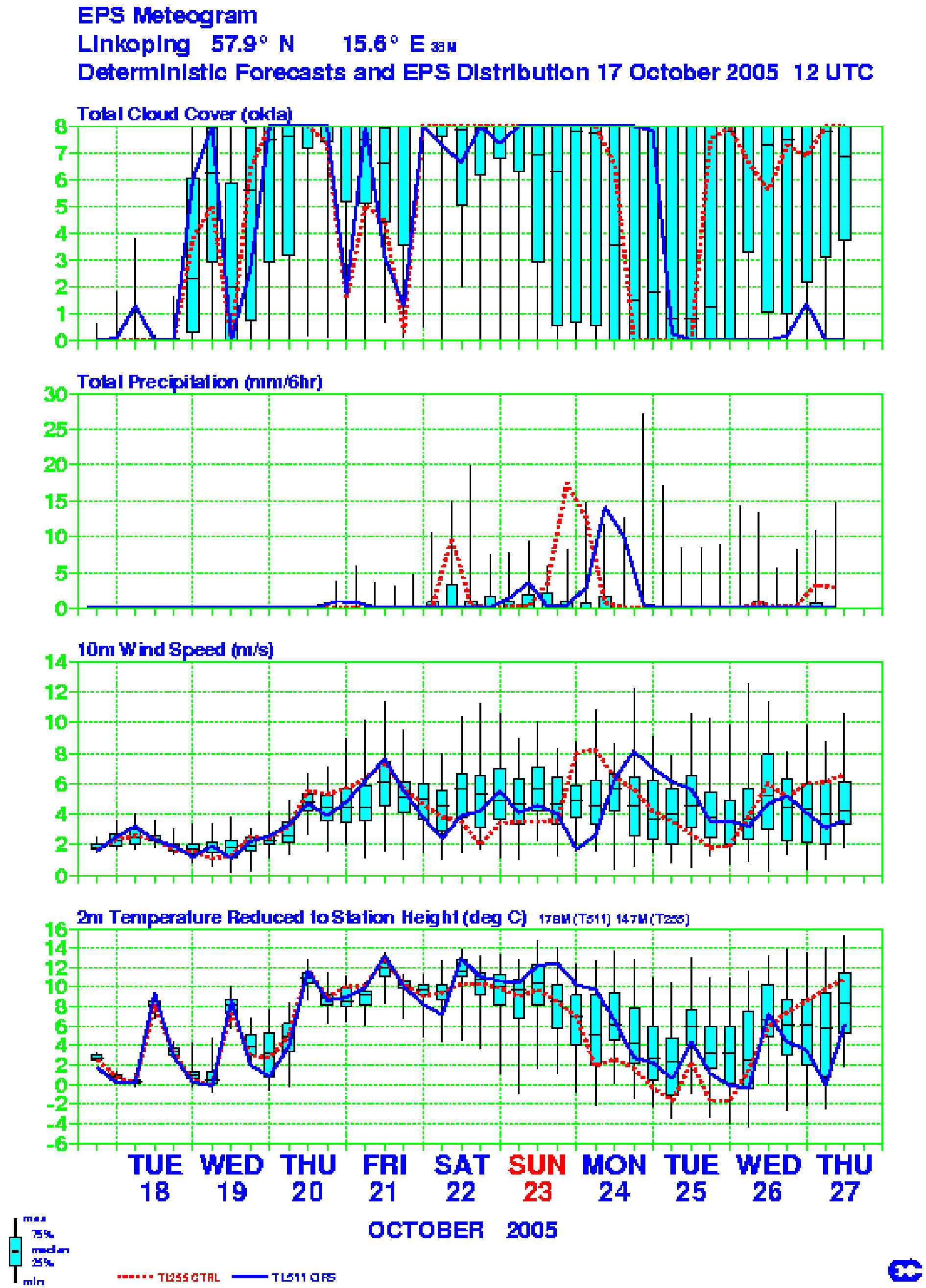
Step forward in time (48 hours, 10 days, …)

Ensemble forecast

Estimate uncertanty by runing many (50-100) forcasts from perturbed analysis

A 10-day ensemble forecast for Linköping

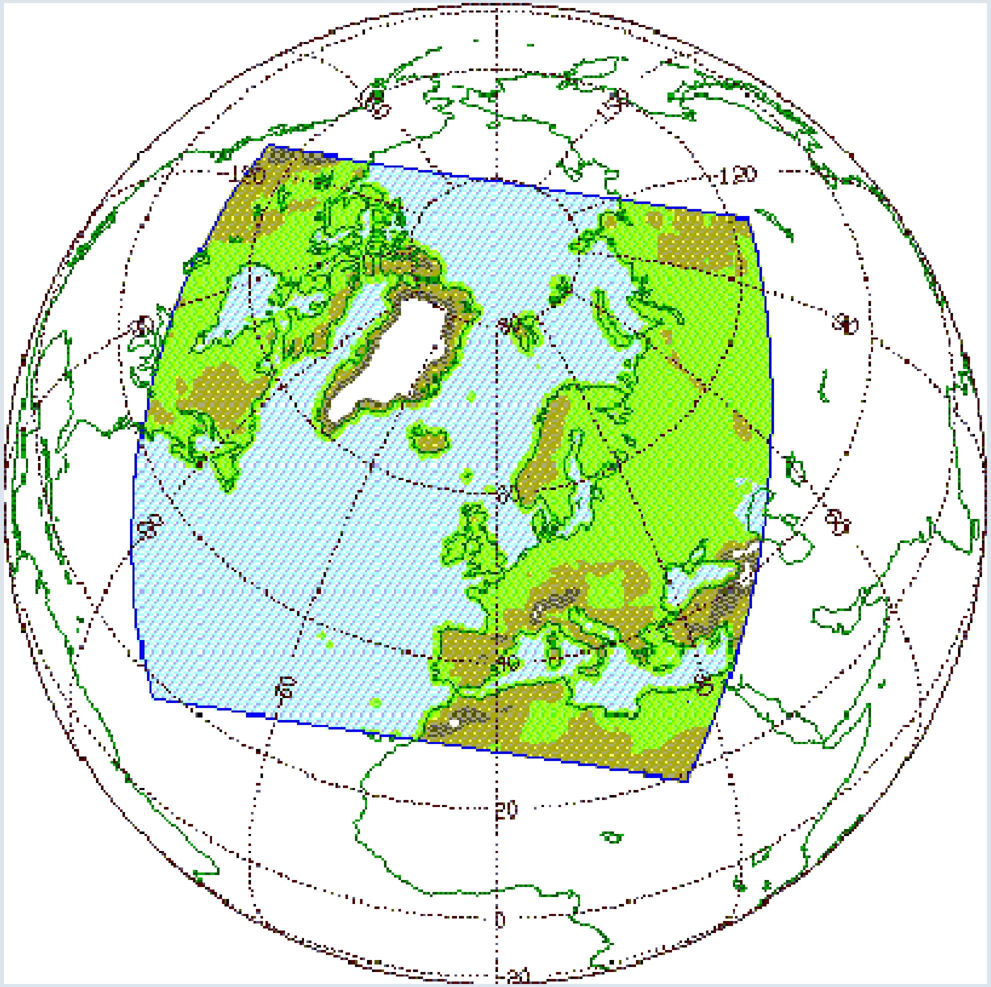
- Blue line is unperturbed high resolution forecast
- Dotted red is unperturbed reduced resolution forecast
- Bars indicate center 50% of 100 perturbed forecasts at reduced resolution



SMHI

HIRLAM at SMHI

- 48-hour 22 km resolution forecast on a limited domain
  - Boundaries from global IFS forecast at 40km
- Also 11 km HIRLAM forecast on a smaller domain
- 40 minutes elapsed on 32 processor of a Linux cluster
  - Dual Intel Xeon 3.2 GHz
  - Infiniband
  - More info in Torgny Faxén’s talk tomorrow!



Codes: IFS, ALADIN, HIRLAM, ALADIN

- **IFS – Integrated Forecast System (ECMWF)**
  - **Global, Spectral, 2D decomposition, 4D-VAR**
- **ALADIN - Aire Limitée Adaptation Dynamique développement InterNaltional**
  - **Shares codebase with ARPEGE, the Météo-France version of IFS**
  - **Limited area, Spectral, 2D decomposition, 3D-VAR**
  - **Future: AROME at 2-3 km scale**
- **HIRLAM – High Resolution Limited Area Model**
  - **Limited area, Finite difference, 2D decomposition**
- **HIRVDA – HIRlam Variational Data Assimilation**
  - **Limited area, Spectral, 1D decomposition, 3D-VAR, (and soon 4D-VAR ☺)**

Numerics

Longer time steps made possible by

- **Semi-implicit time integration**
  - **Advance fast linear modes implitly and slower non-linear modes explicitly**
  - **A Helmholz equation has to be solved**
    - **In HIRLAM by direct FFT + tri-diagonal method**
    - **Spectral models do it easily in Fourier space**
  - **Implications for domain decomposition!**

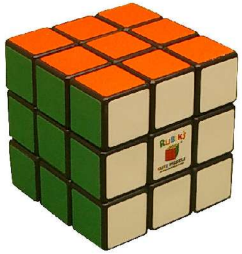
- **Semi-Lagrangian advection**
  - **Wide halo zones**

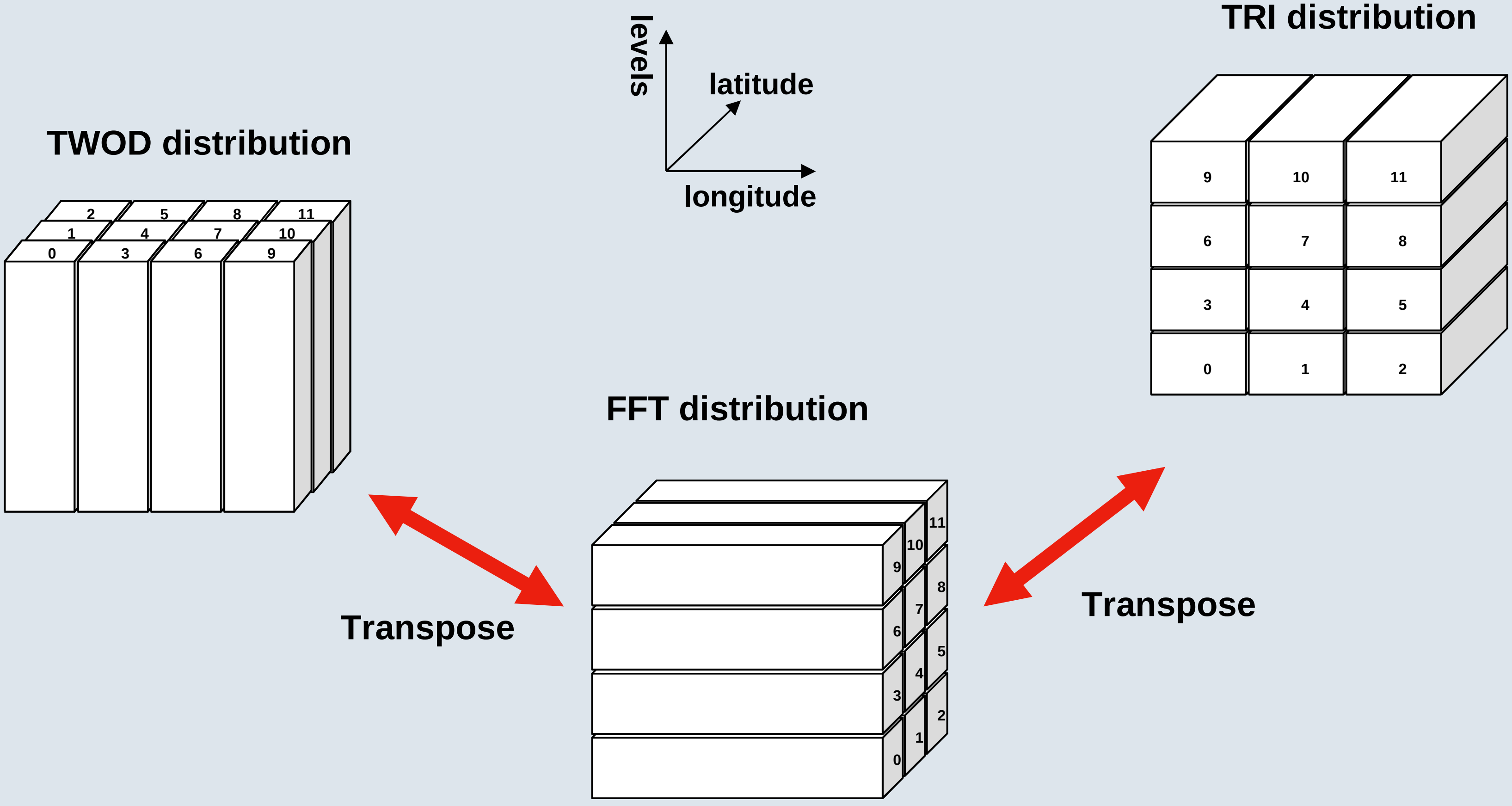
How should we partition the grid?

- Example: HIRLAM C22 grid   (nx = 306, ny = 306, nlev = 40)
  - Many complex interactions in vertical (the ”physics”).
    - Decomposing the vertical would mean frequent interprocessor communication.
  - Helmholtz solver
    - FFT part prefers nondecomposed longitudes
    - Tridiagonal solver partpreferes nondecomposed latitudes

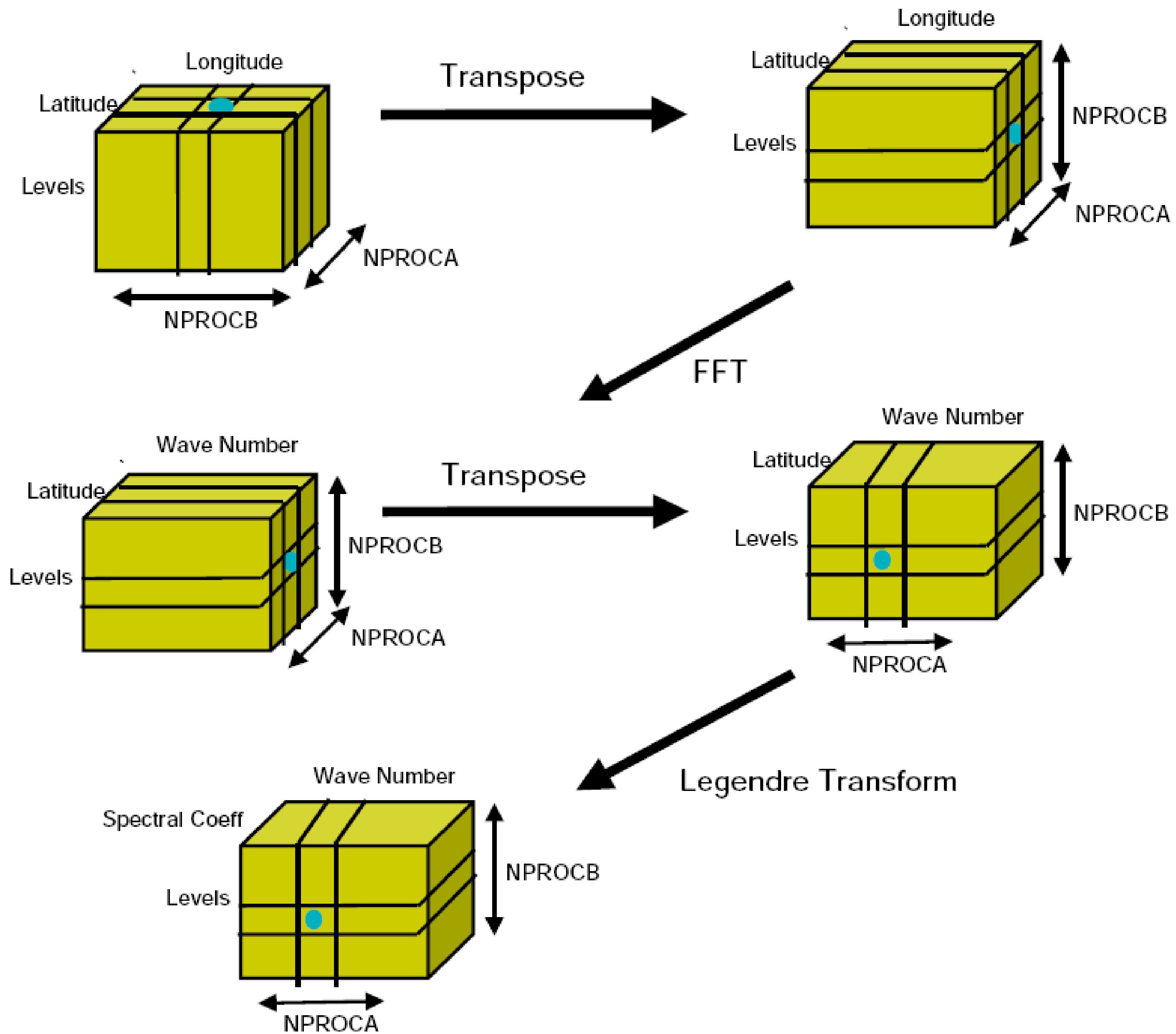
- Similar for spectal models (IFS, ALADIN & HIRVDA)
  - Transforming from physical space to spectral space means
    - FFTs in both longitiudes and latitudes
    - And physics in vertical

Grid partitioning in HIRLAM
(Jan Boerhout, NEC)



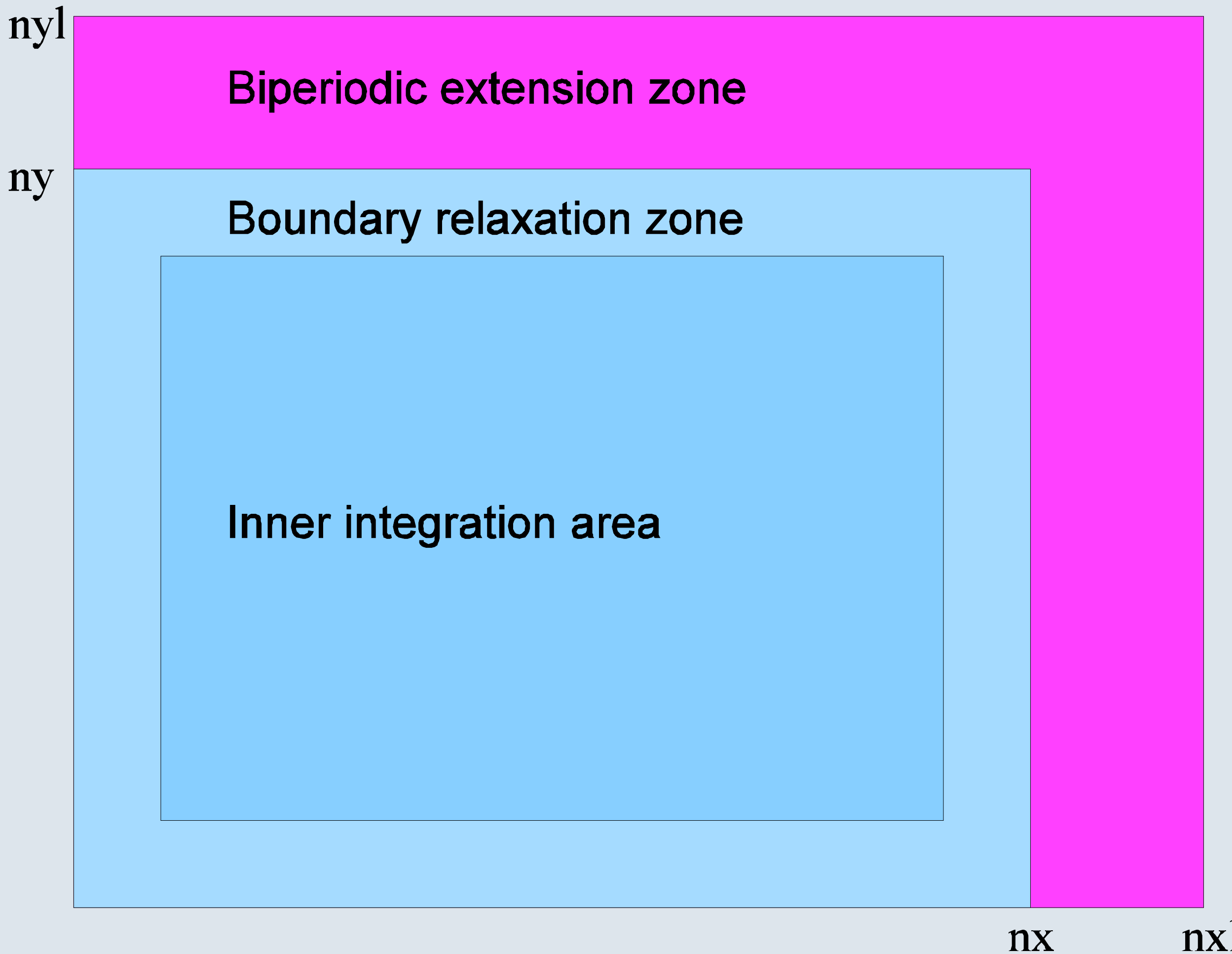


Transforms and transposes in IFS / ALADIN

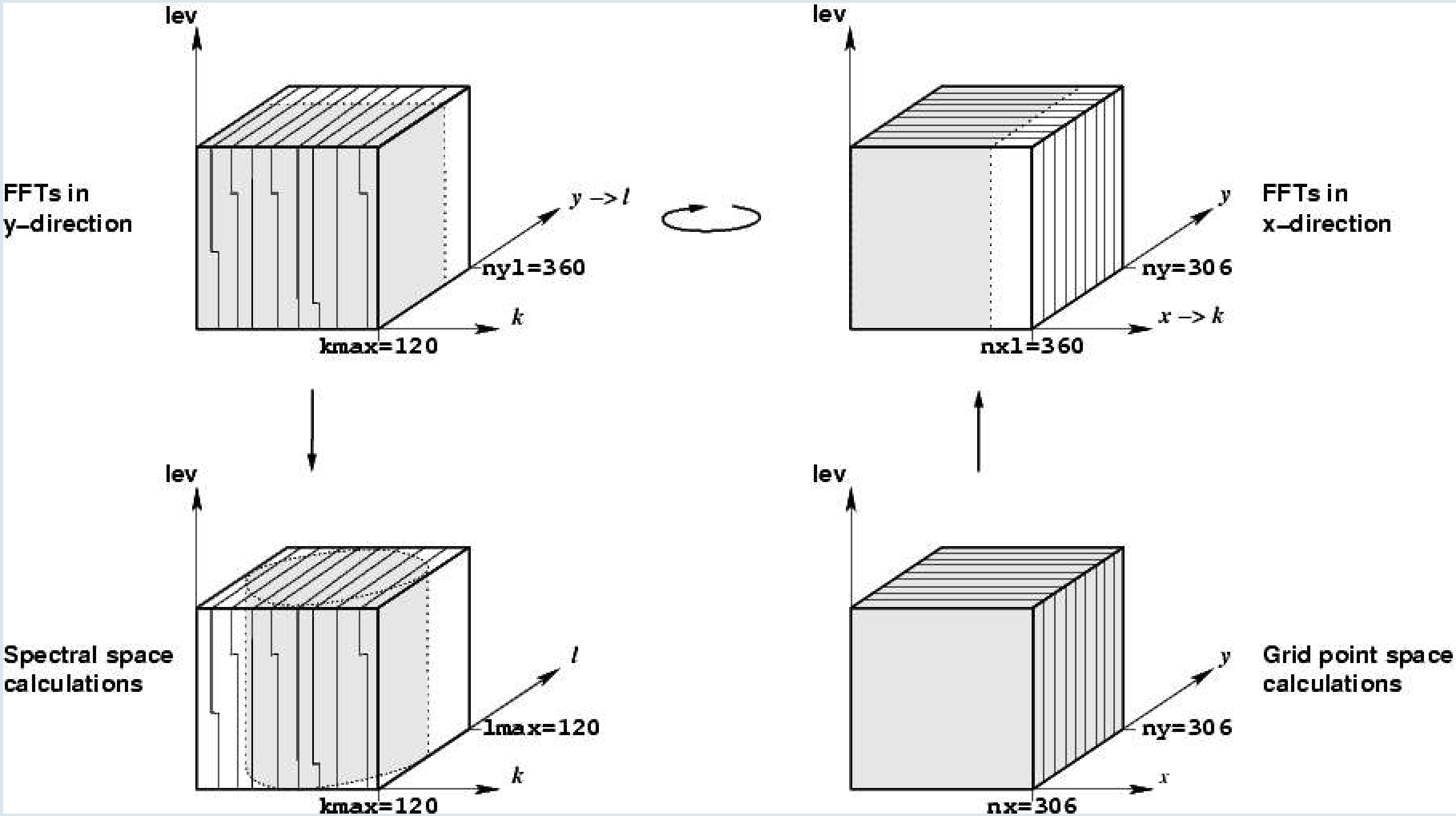


Spectral methods in limited area models HIRVDA / ALADIN

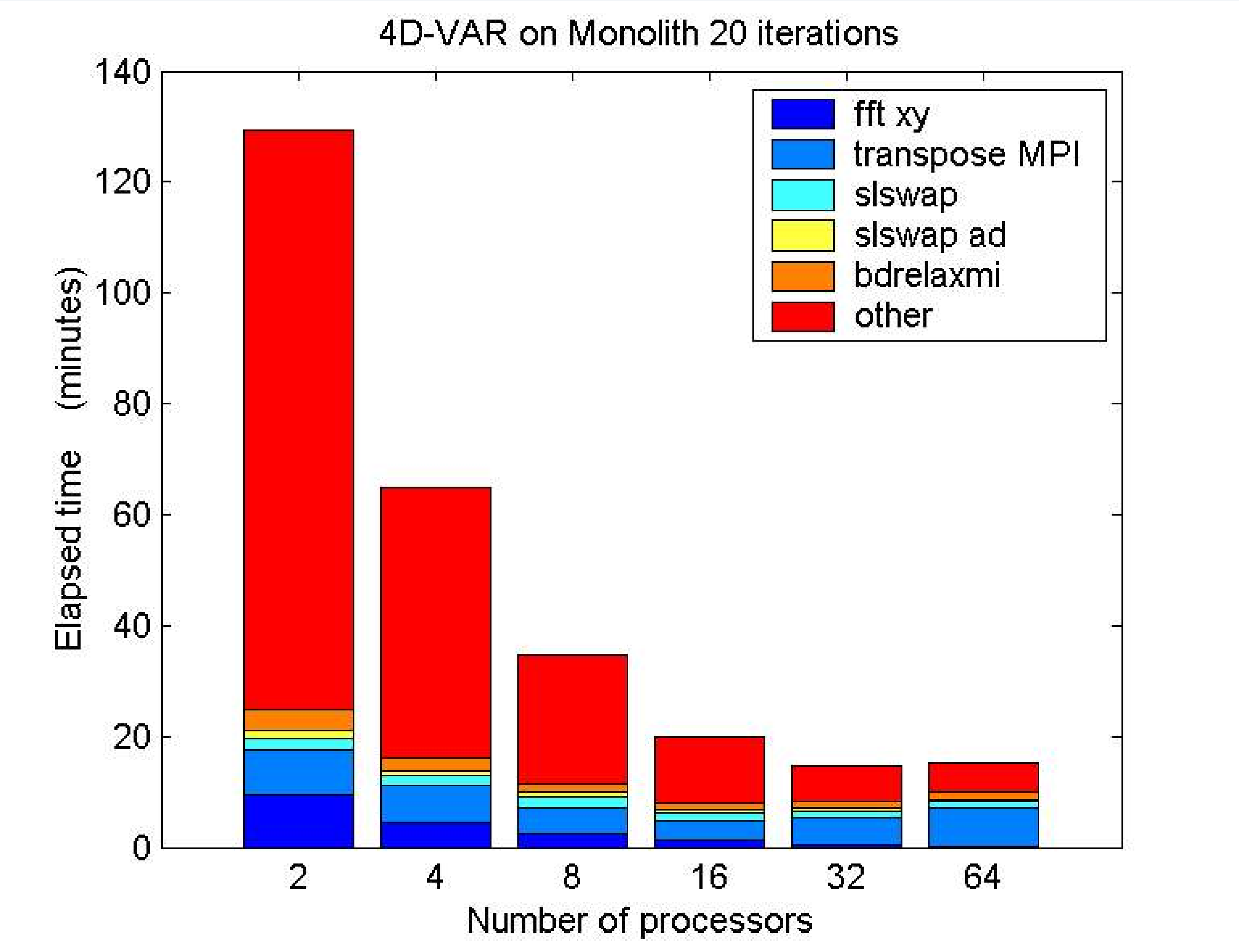
- HIRVDA C22 domain
  - nx & ny = 306
- Extension zone
  - nxl & nyl = 360
- Spectral space
  - kmax & lmax = 120



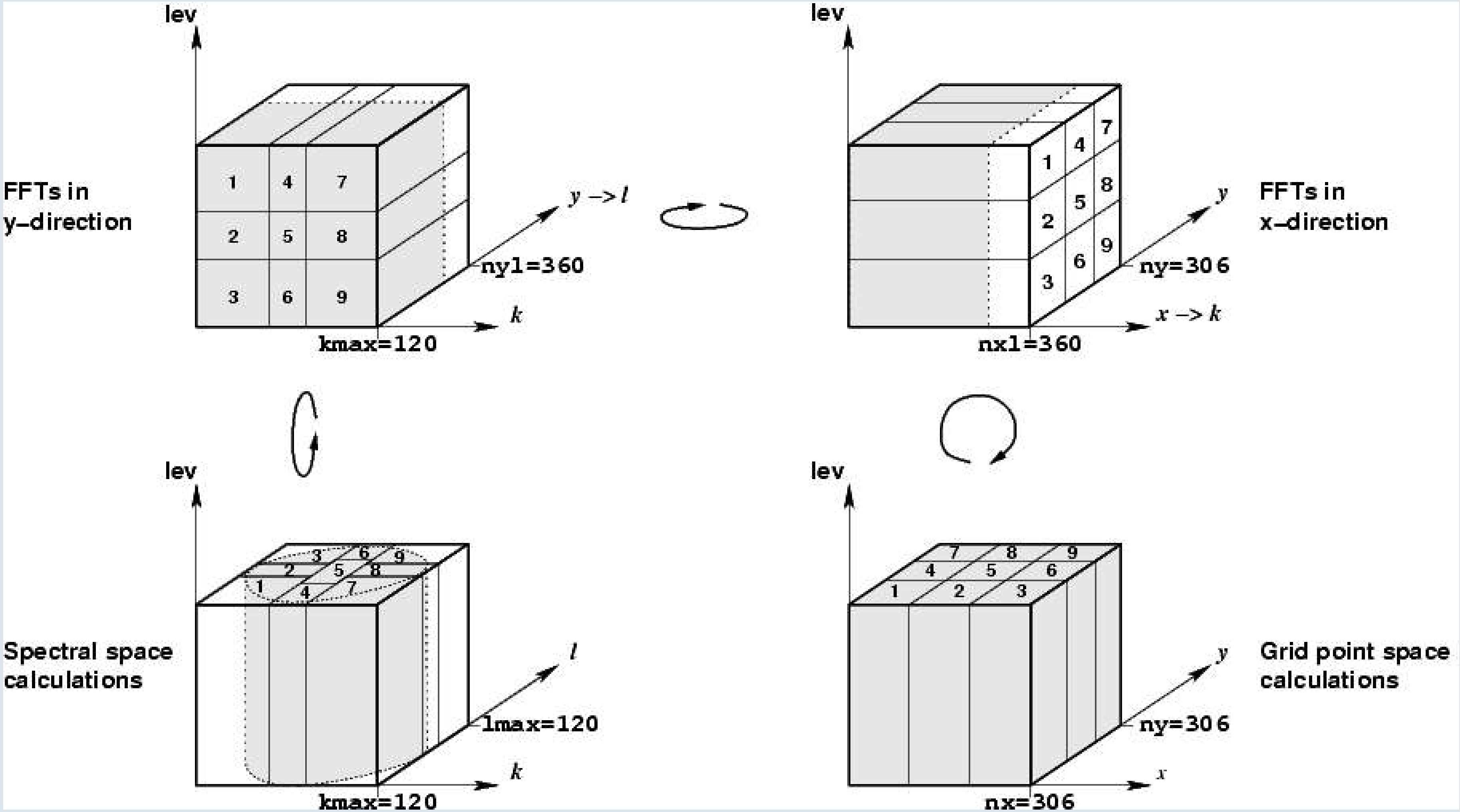
Transposes in HIRVDA (spectral HIRLAM)
1D decomposition



HIRVDA timings



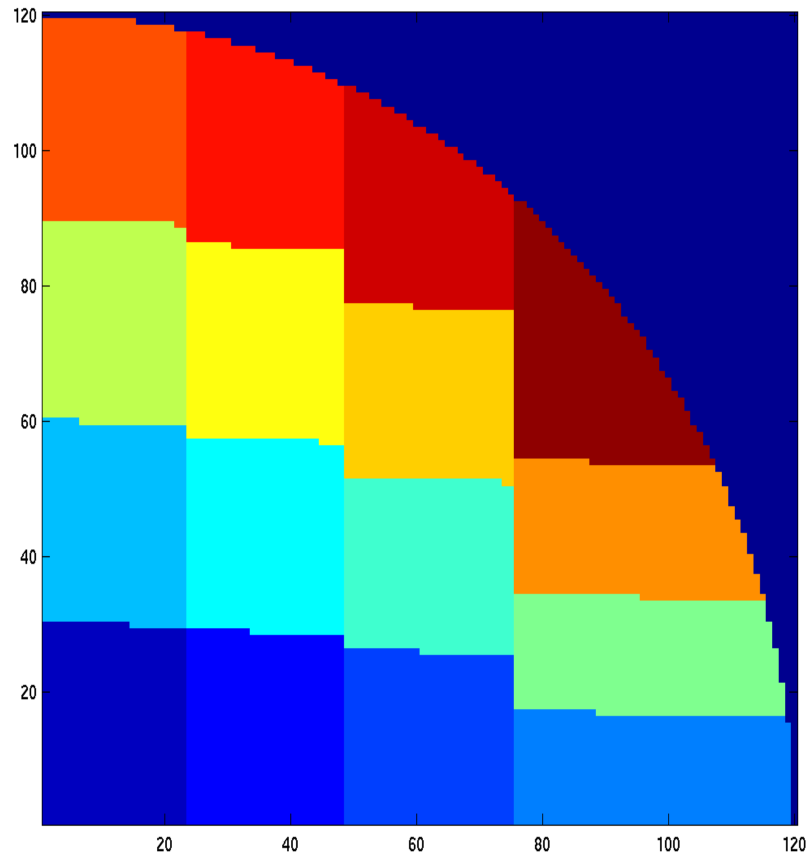
Transposes with 2D partitioning



Load balancing in spectral space

- Isotropic representation in spectral space requrires an ellipic truncation

- By accepting an unbalanced y-direction FFT, spectral space can be load balanced



Number of messages

1D decomposition

n=4 => 24 n=64 => 8064

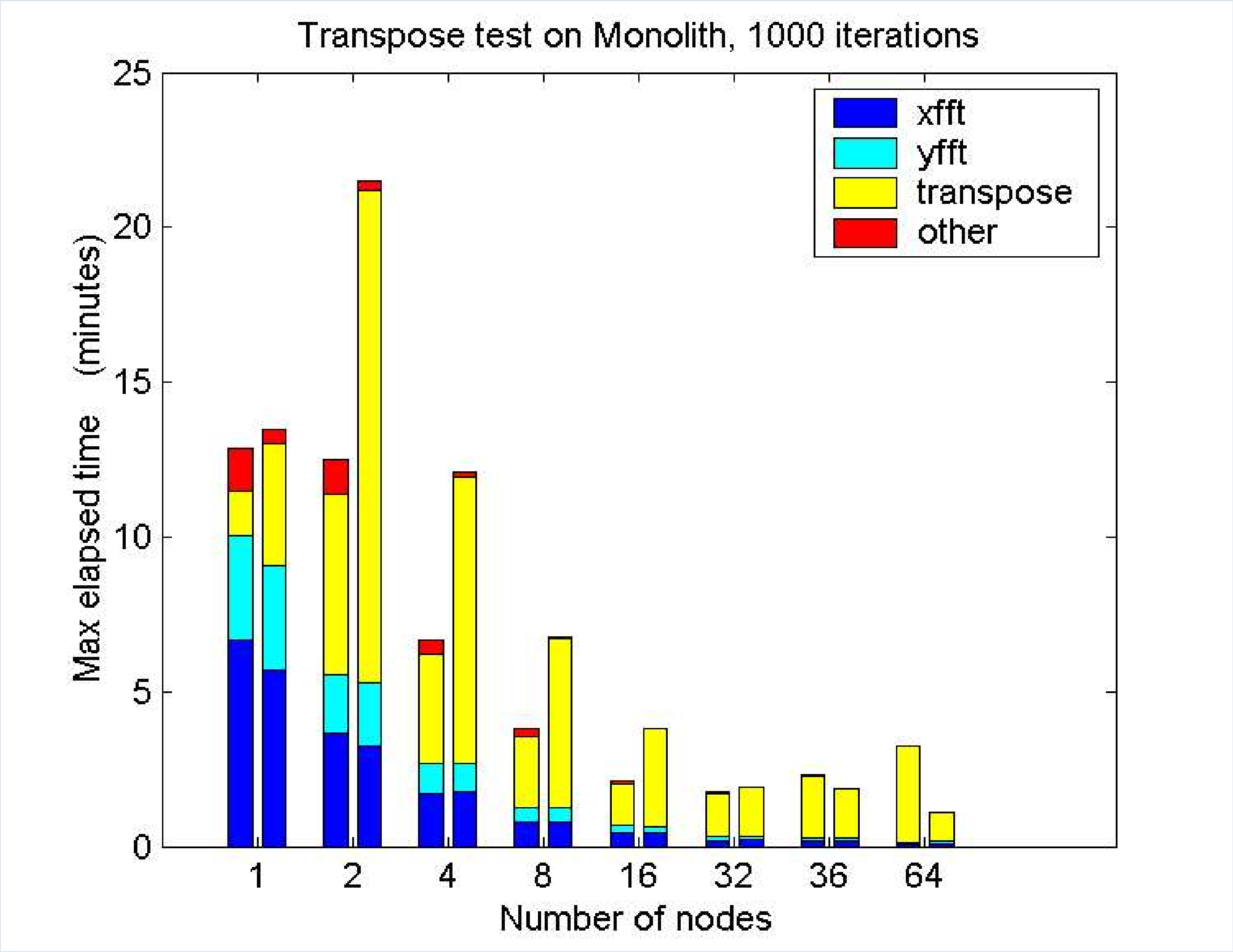
$$2n(n-1)$$

2D decomposition

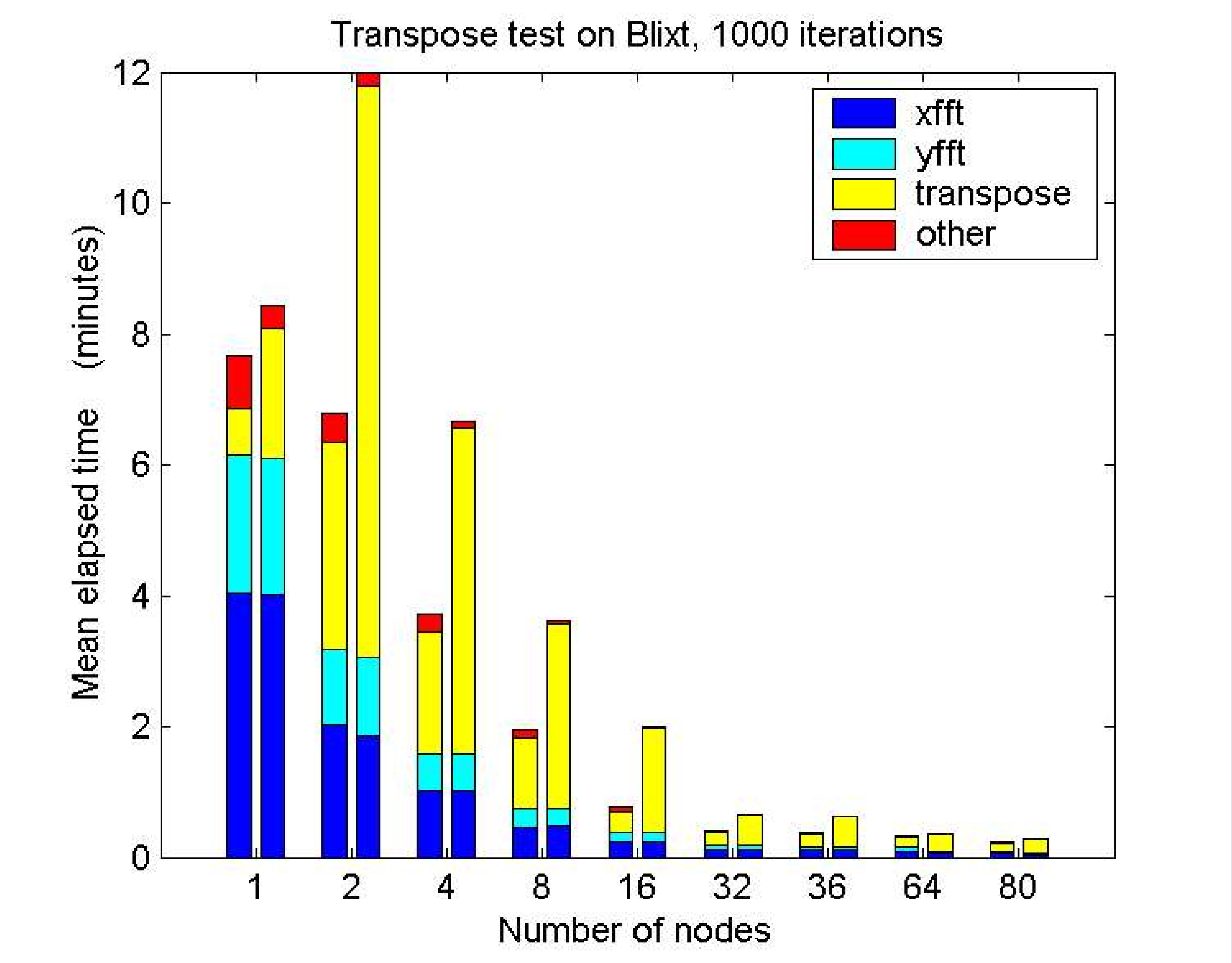
n=4 => 24 n=64 => 2688

$$¿6n(\sqrt{n}-1)$$

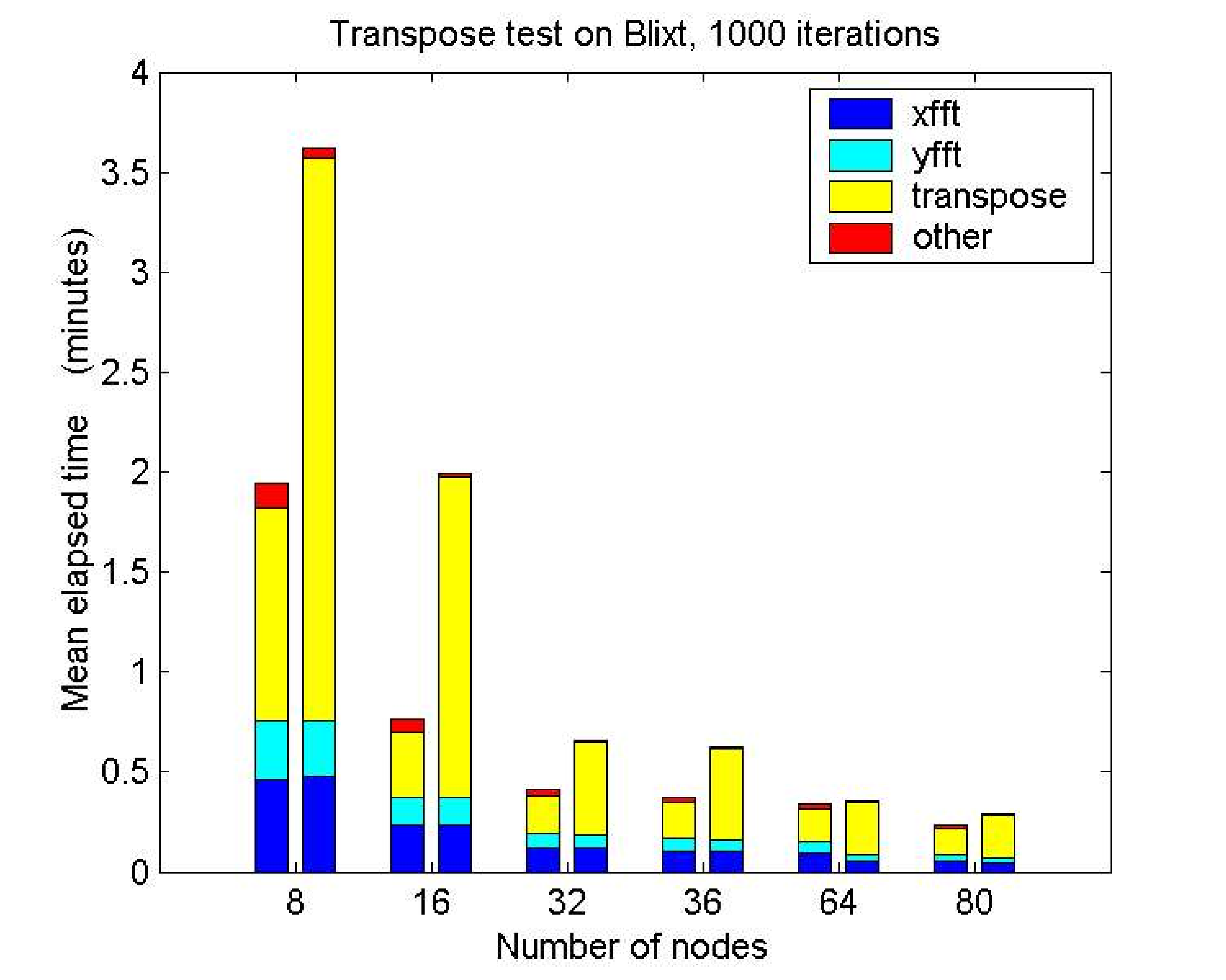
Timings on old cluster (Scali)



Timings on new cluster (Infiniband)

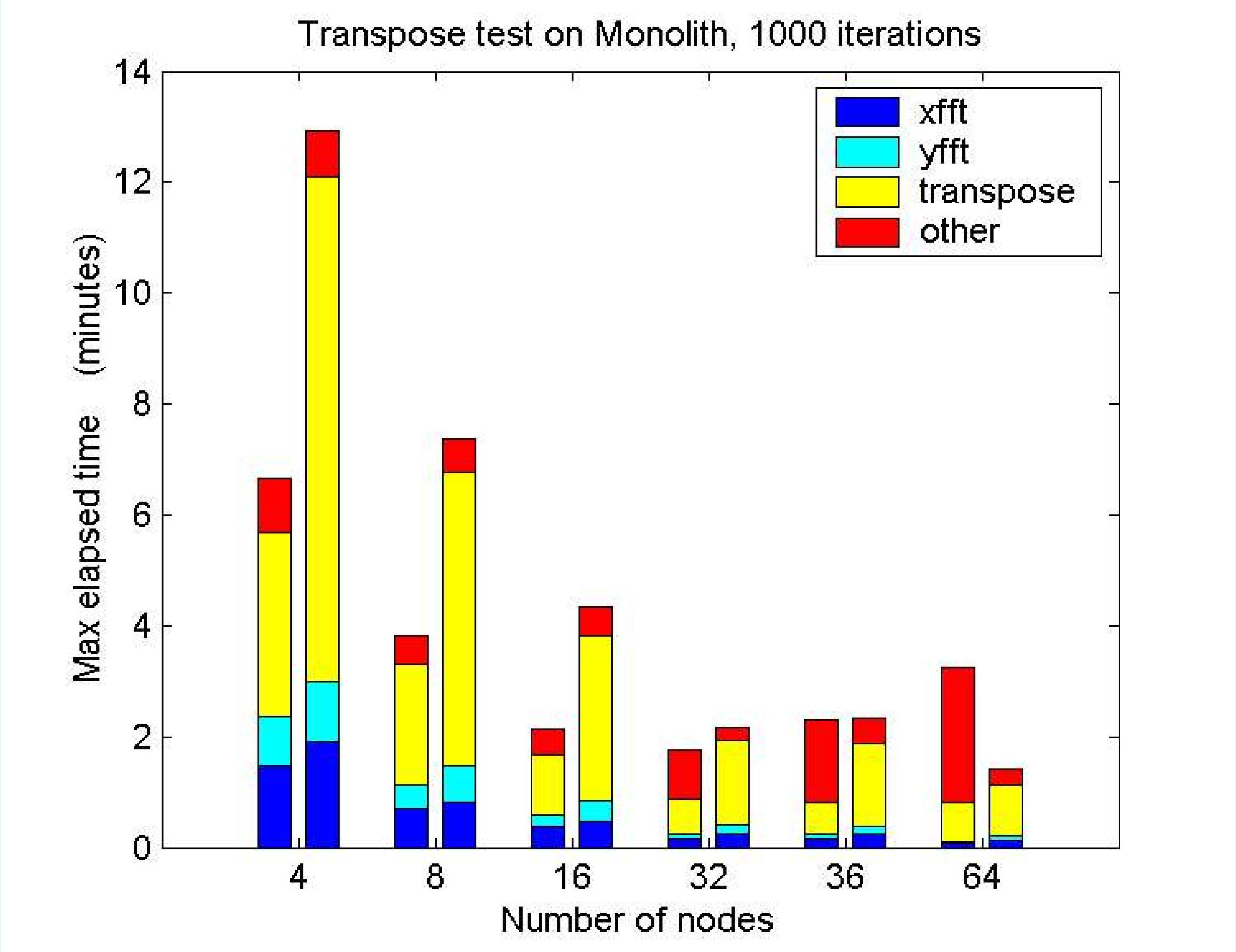


Zoom in…

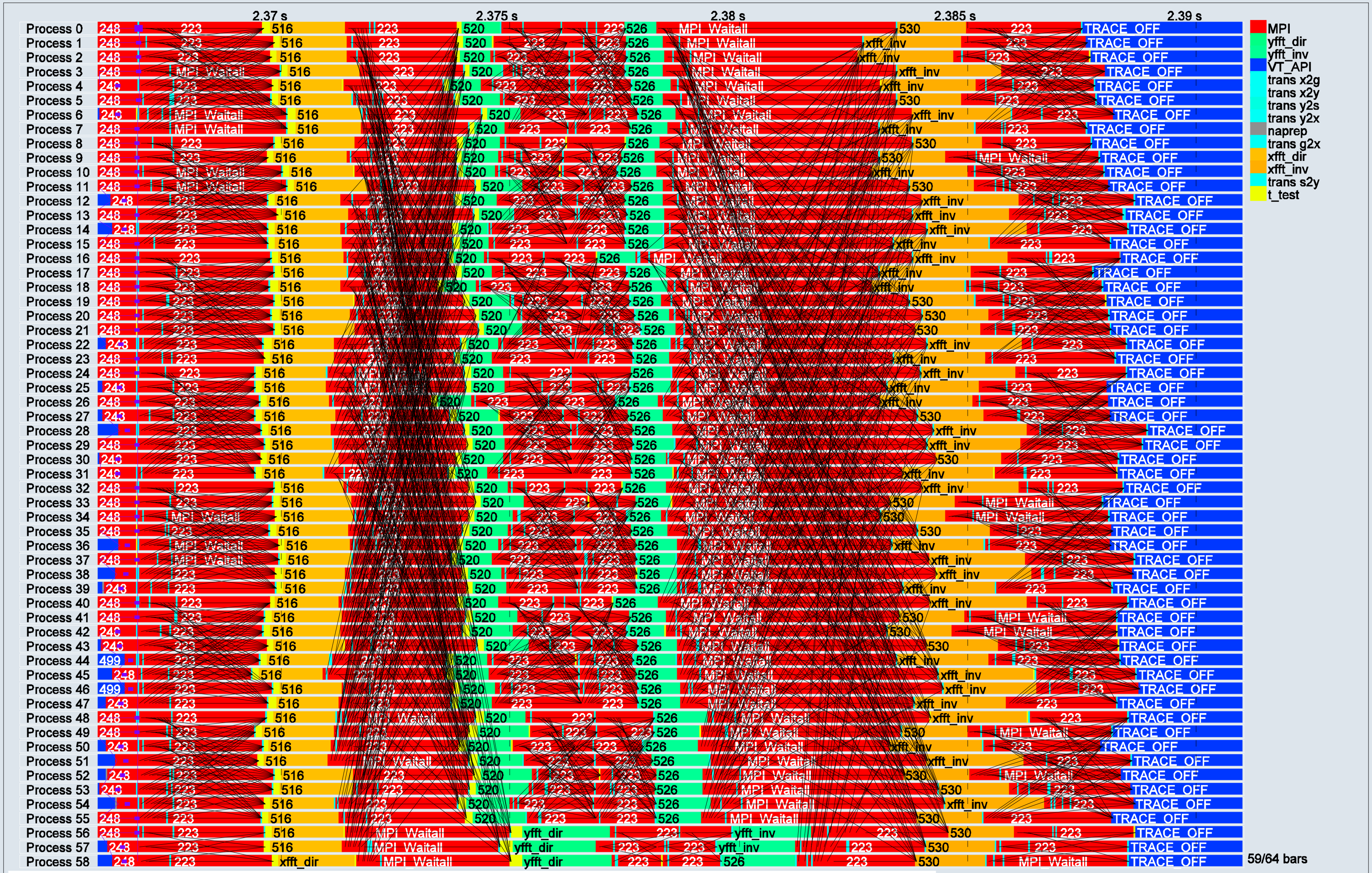


Transpose test on Blixt, 1000 iterations
Mean elapsed time (minutes)
Number of nodes
xfft
yfft
transpose
other
0
0.5
1
1.5
2
2.5
3
3.5
4
8
16
32
36
64
80

Minimum time on old cluster



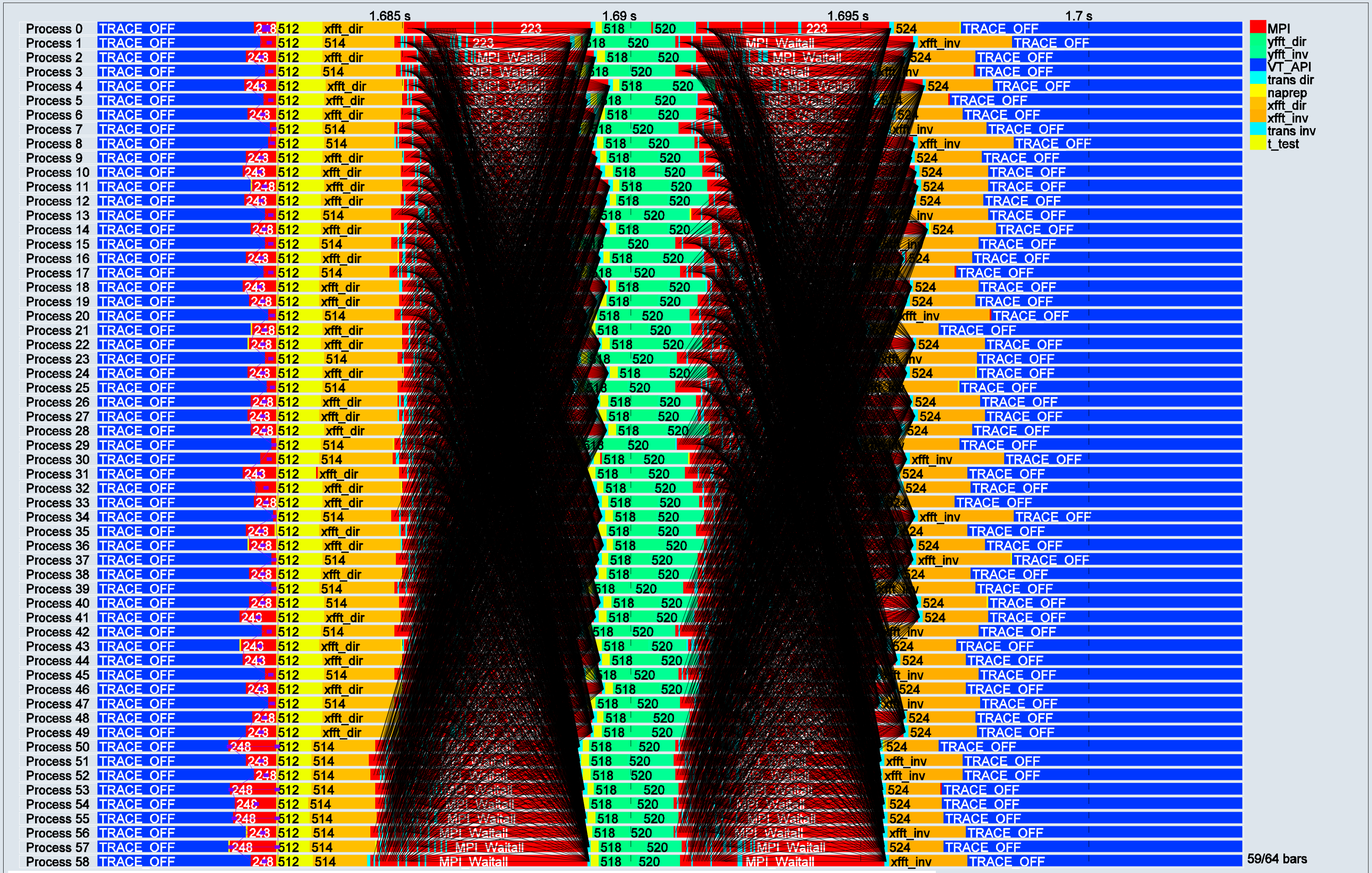
FFT / Transpose timeline
2D decomposition



twod-nonbuff.x-0(mpi:10043@n41).stf (2.366 s - 2.391 s = 25.053 ms)

Printed by Intel(R) Trac… Analyzer

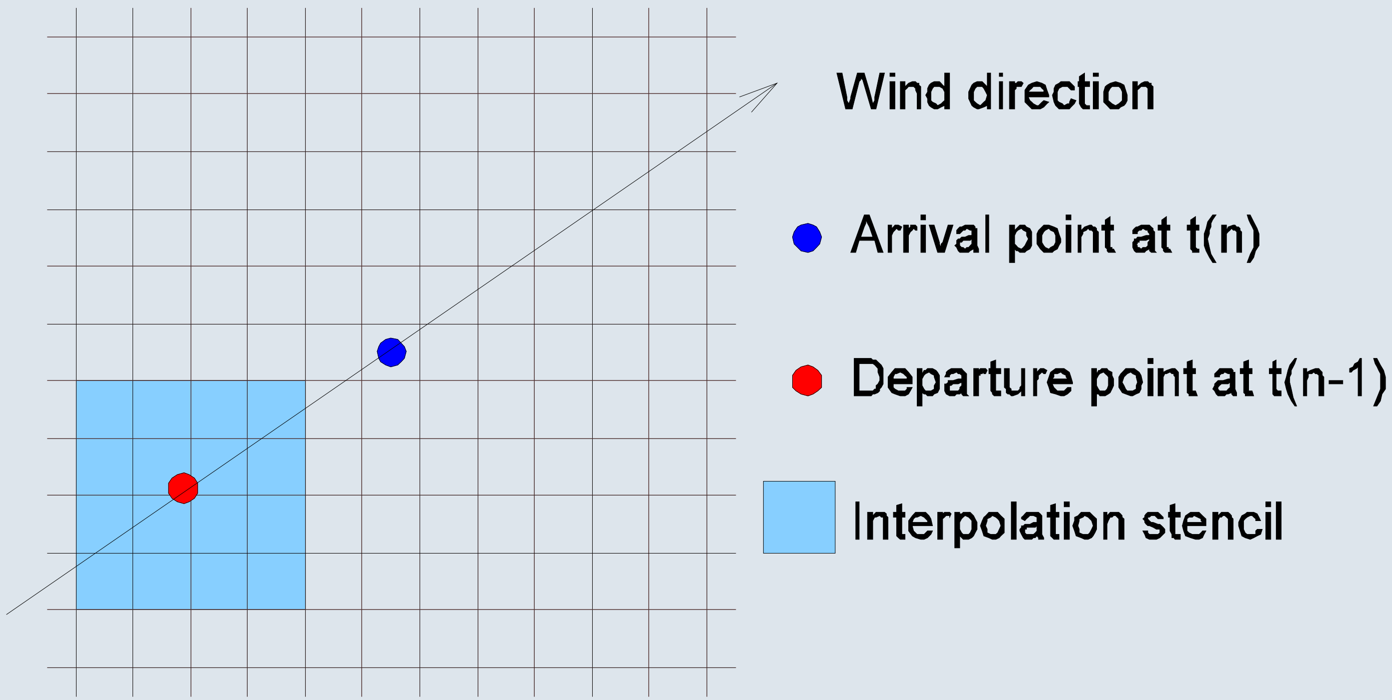
FFT / Transpose timeline
1D decomposition



orig-nonbuff.x-0(mpi:9888@n41).stf (1.678 s - 1.703 s = 24.996 ms)

Printed by Intel(R) Trace Analyzer 4

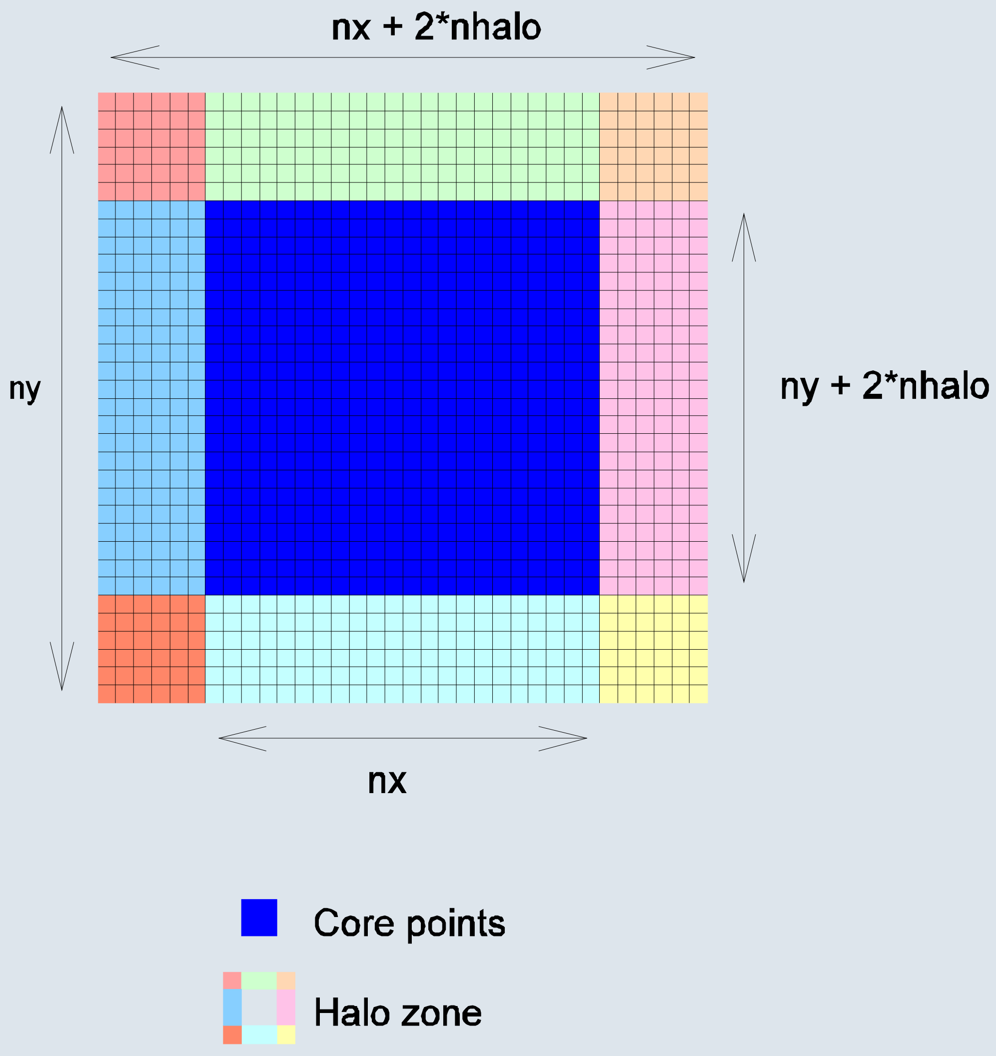
Semi-Lagrangian Advection



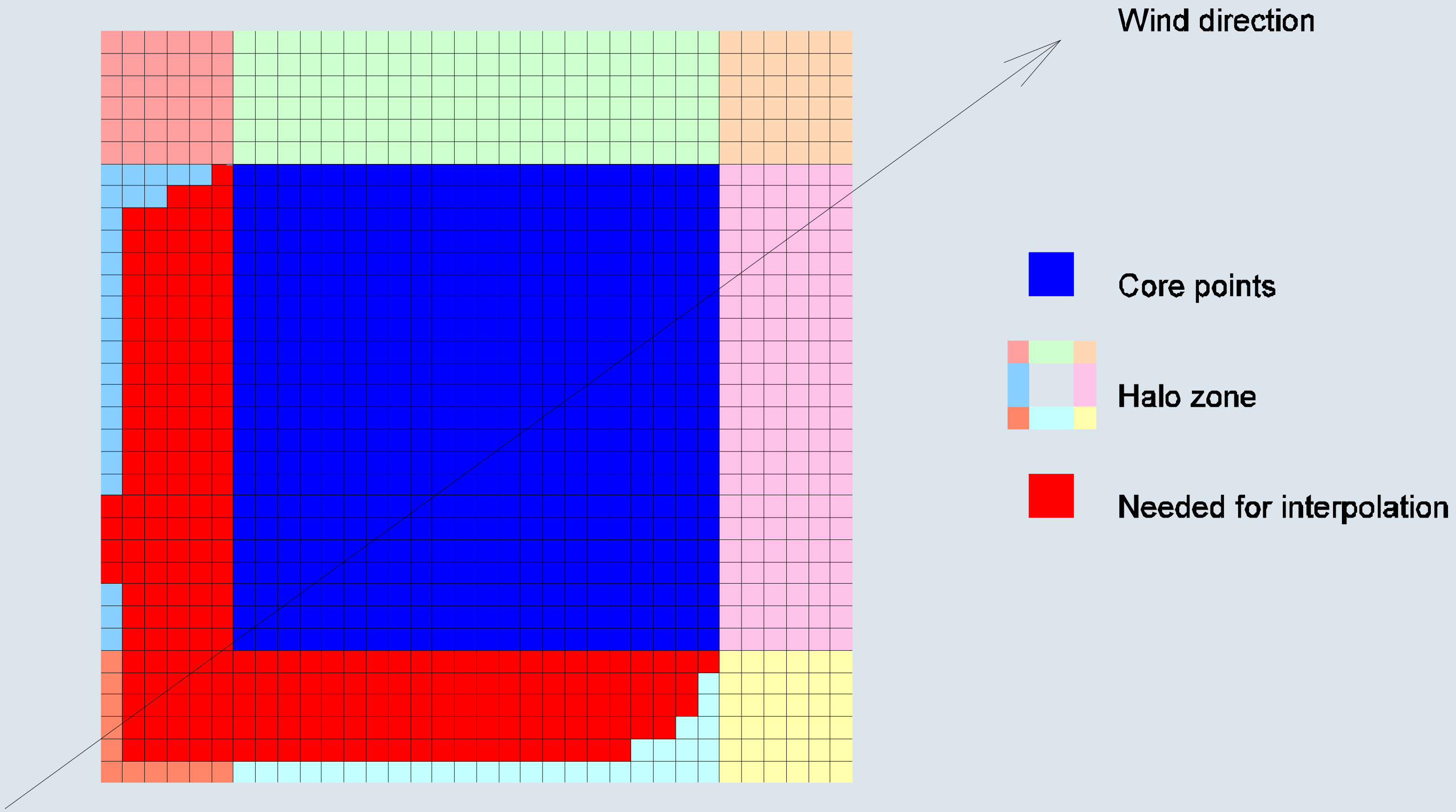
- **Full cubic interpolation in 3D is 32 points (4x4x4)**

Example: The HIRLAM C22 area (306x306 grid at 22 km resolution)

- Max wind speed in jet stream 120 m/s
- Time step 600 s
- => Distance 72 km = 3.3 grid points)
- Add stencil width (2) => nhalo= 6
- With 64 processors partitioned in 8x8:
  - 38x38 core points per processor
  - 50x50 including halo
- Halo area is 73% of core!
- But full halo is not needed everywhere!



IFS & ALADIN – Semi-Lagrangian advection
Requesting halo points ’on-demand’

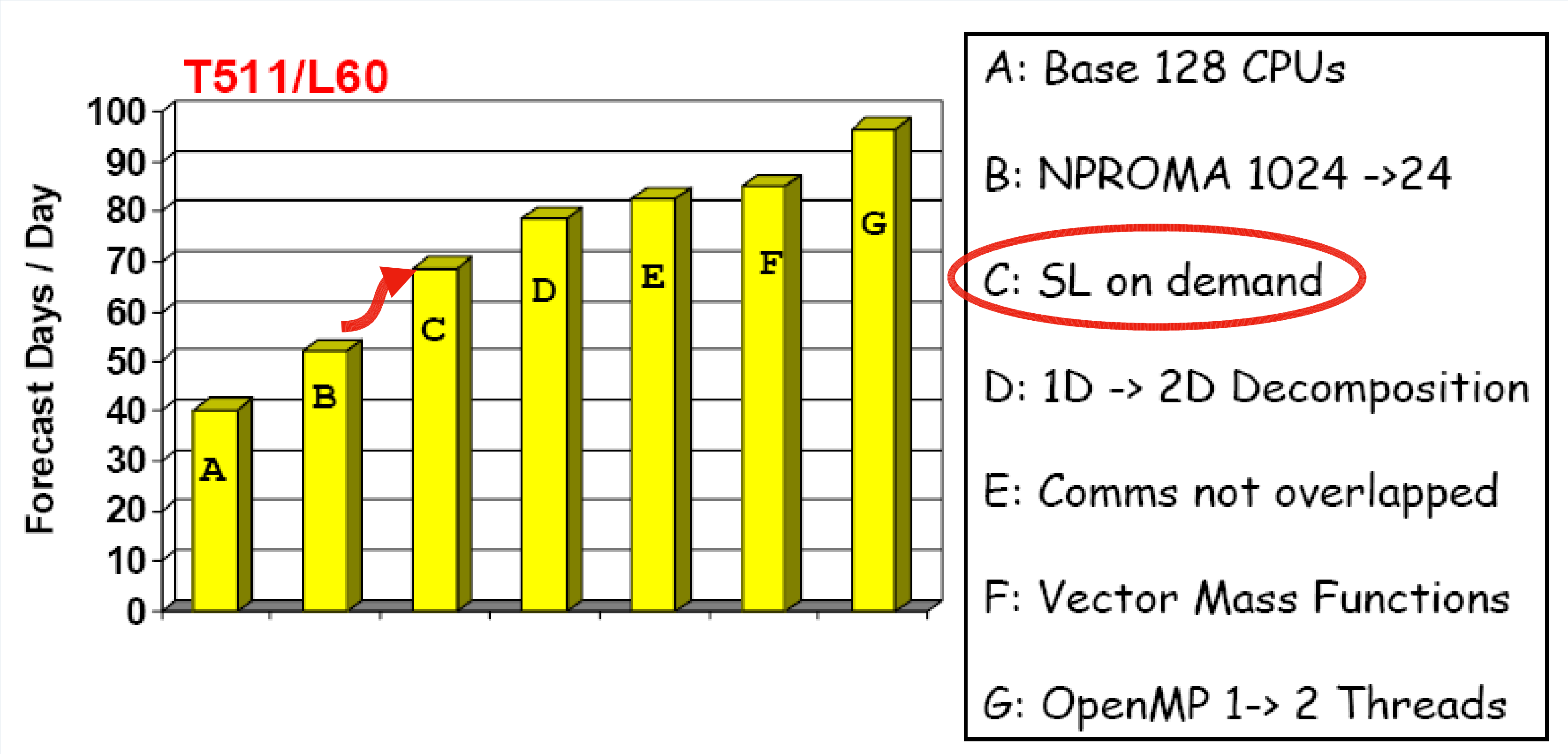


SMHI

’On-demand’ algorithm

1. Exchange full halo for wind components (u,v & w)
2. Calculate departure points
3. Determine halo-points needed for interpolation
4. Send list of halo points to surrounding PE’s
5. Surrounding PE’s send points requested

Effects on various optimizations on IFS performance



Moving from Fujitsu VPP (vector machine) to IBM SP (cluster).

Figure from Debora Salmond (ECMWF).

SMHI

Conclusion

- Meteorology and climate sciences provide plenty of fun problems for somebody interested in computational methods and parallelization. Also:
  - Load balancing observations in data assimilation
  - Overlapping I/O with computation

- HIRLAM & HIRVDA will get ’on-demand’ slswap soon