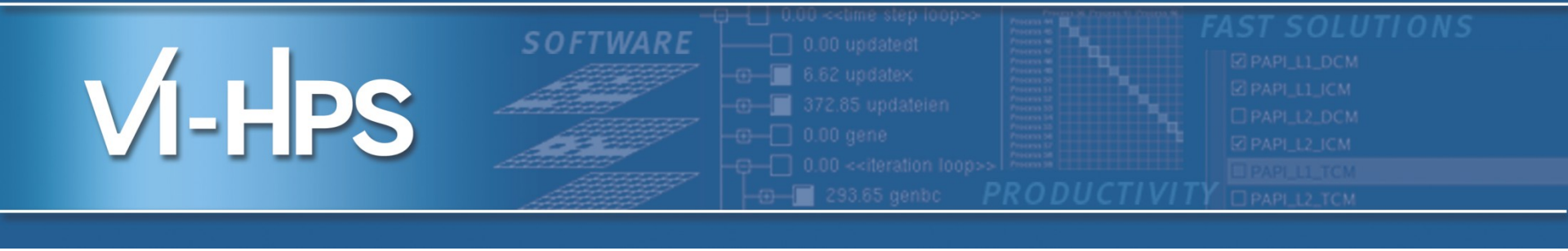


PFLOTRAN Case Study

Brian Wylie & Markus Geimer, Jülich Supercomputing Centre
Tobias Hilbrich & Matthias Lieber, Technische Universität Dresden
Chee Wai Lee & Wyatt Spear, University of Oregon

supported by the PFLOTRAN & VI-HPS development teams

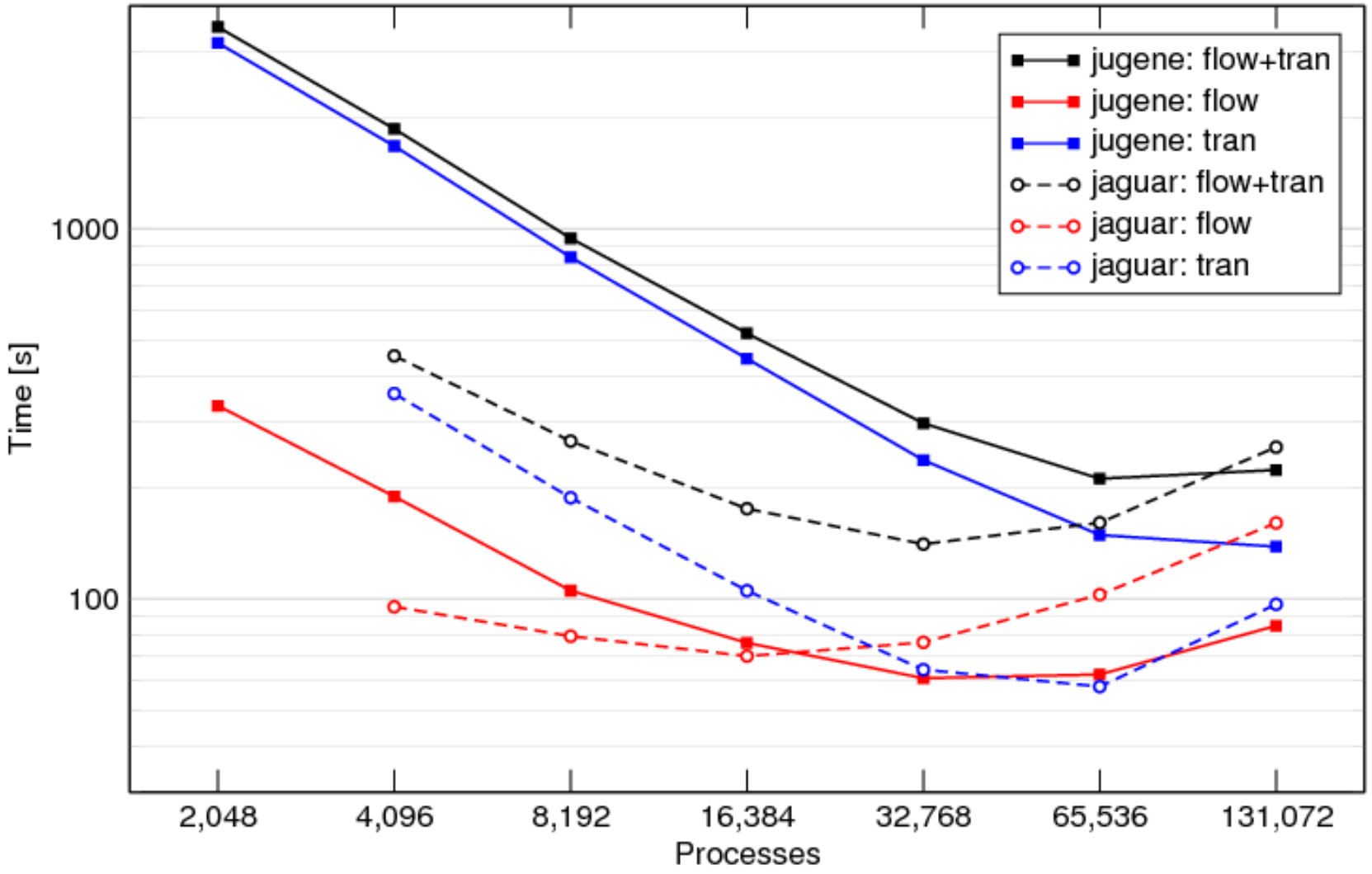
Outline

- PFLOTRAN application
  - characteristics, scalability (BG/P & XT5)
- Scalasca
  - summary, summary+PAPI & trace analyses
- Vampir
  - timeline visualization, clustering, communication matrix, etc.
- TAU
  - PDT selective instrumentation
  - ParaProf experiment manager, PerfExplorer, 2D/3D profiles, histograms, callgraph, etc.

PFLOTRAN

- 3D reservoir simulator developed by LANL/ORNL/PNNL
  - http://ees.lanl.gov/source/orgs/ees/pflotran/
  - approx. 80 thousand lines of Fortran90, combining
    - PFLOW non-isothermal, multi-phase groundwater flow solver
    - PTRAN reactive, multi-component contaminant transport solver
  - employs PETSc, LAPACK, BLAS & HDF5 I/O libraries
    - 87 PFLOTRAN source files (72 modules) + 789 PETSc
  - run with “2B” input dataset for 10 timesteps
    - most of run time in initialization phase, typically amortized
    - uses 3-dimensional (non-MPI) PETSc Cartesian grid
  - alternating FLOW & TRAN(sport) steps in each timestep
    - scaled on Jugene BG/P to 64k, Jaguar XT5 to 32k
    - TRAN(sport) step scaled much better than flow step
    - FLOW step generally faster, but crossover at larger scale

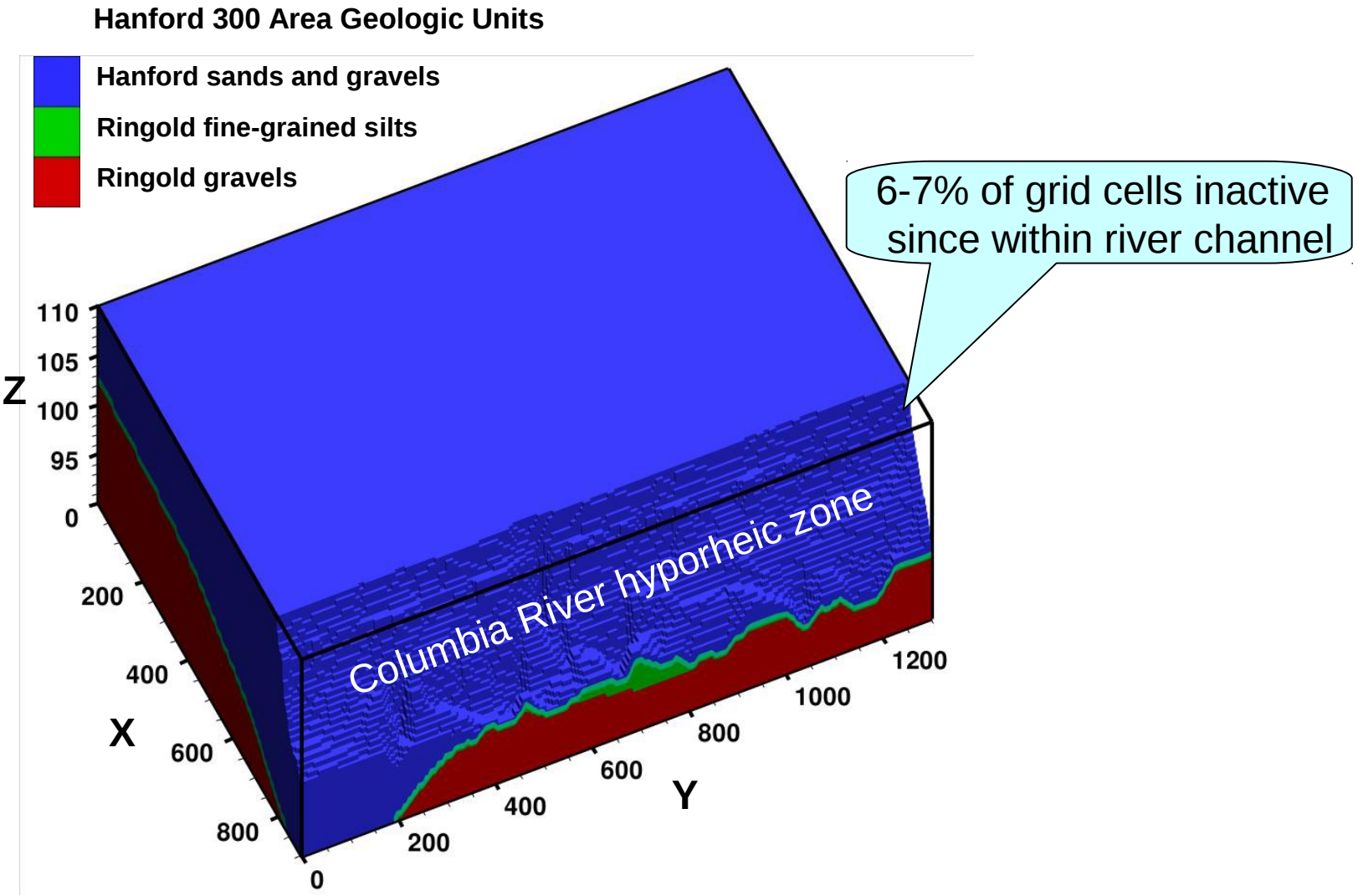
PFLOTRAN (2B.10ts) scalability



Analysis challange

- Demonstrate performance measurement and analysis of PFLOTRAN using “2B” problem dataset and more than 10 thousand MPI processes
  - IBM BG/P (jugene.fz-juelich.de)
  - Cray XT5 (jaguar.nccs.ornl.gov)
- Challenge issued for Dagstuhl seminar 10181 on “Program development for extreme-scale computing” (3-7 May 2010)
  - two months notice with download/build/run instructions
  - http://www.dagstuhl.de/10181

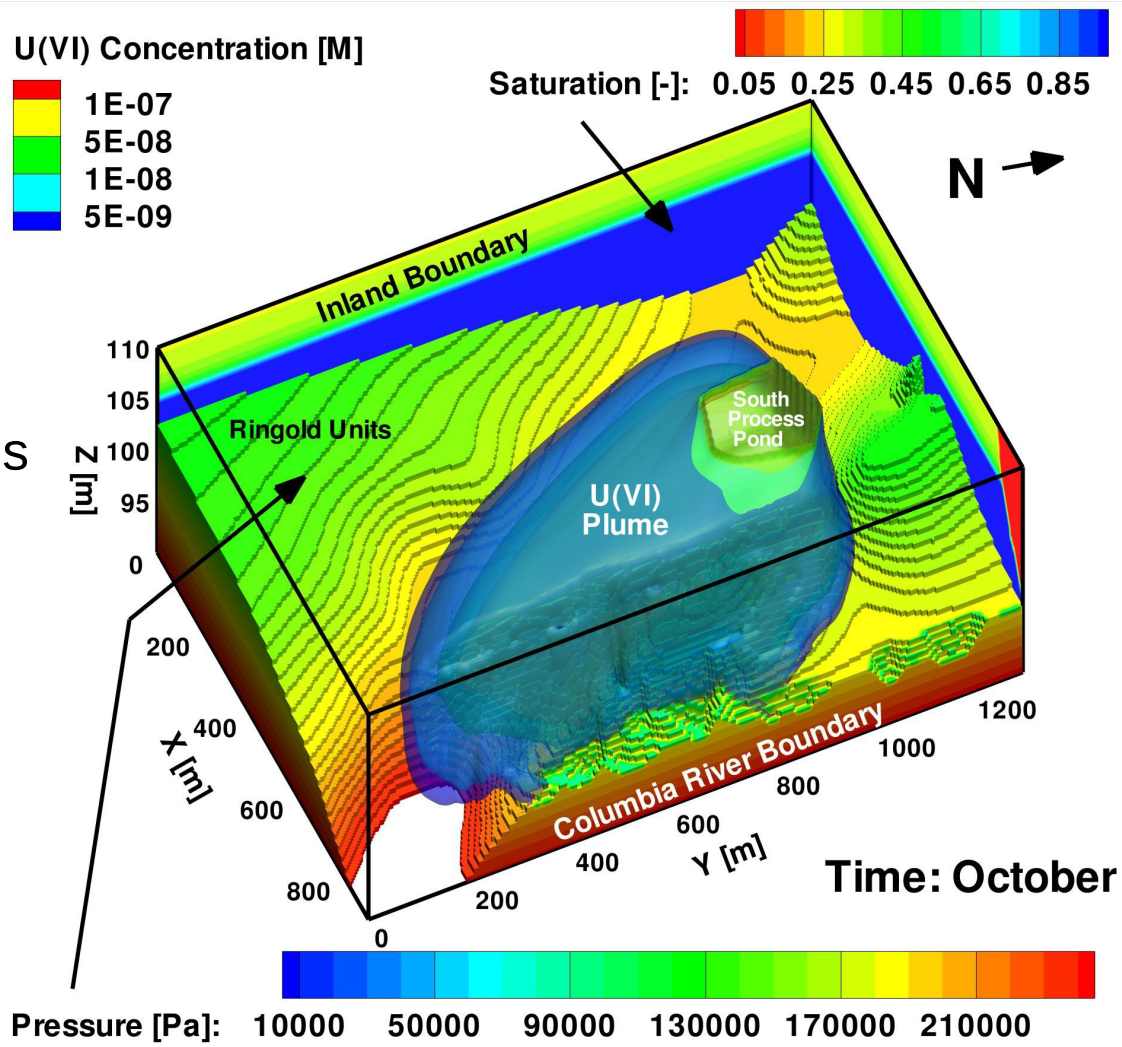
PFLOTRAN “2B” test case



Slide courtesy of G. Hammond (PNNL)

Simulation of U(VI) Migration

- Hanford 300 area
- 1-year simulation:
  - 900x1300x20m
  - $\Delta x/\Delta y$ = 5 m
  - 1.87M grid cells
  - 15 chemical species
  - 28M DoF total
- 1-year simulation:
  - Δt = 1 hour
  - 5-10 hour runtime on Cray XT5 (4k cores)

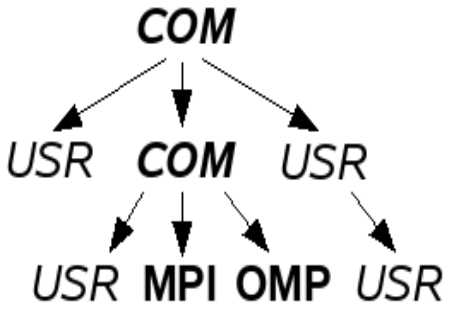


Slide courtesy of G. Hammond (PNNL)

Usage of Scalasca

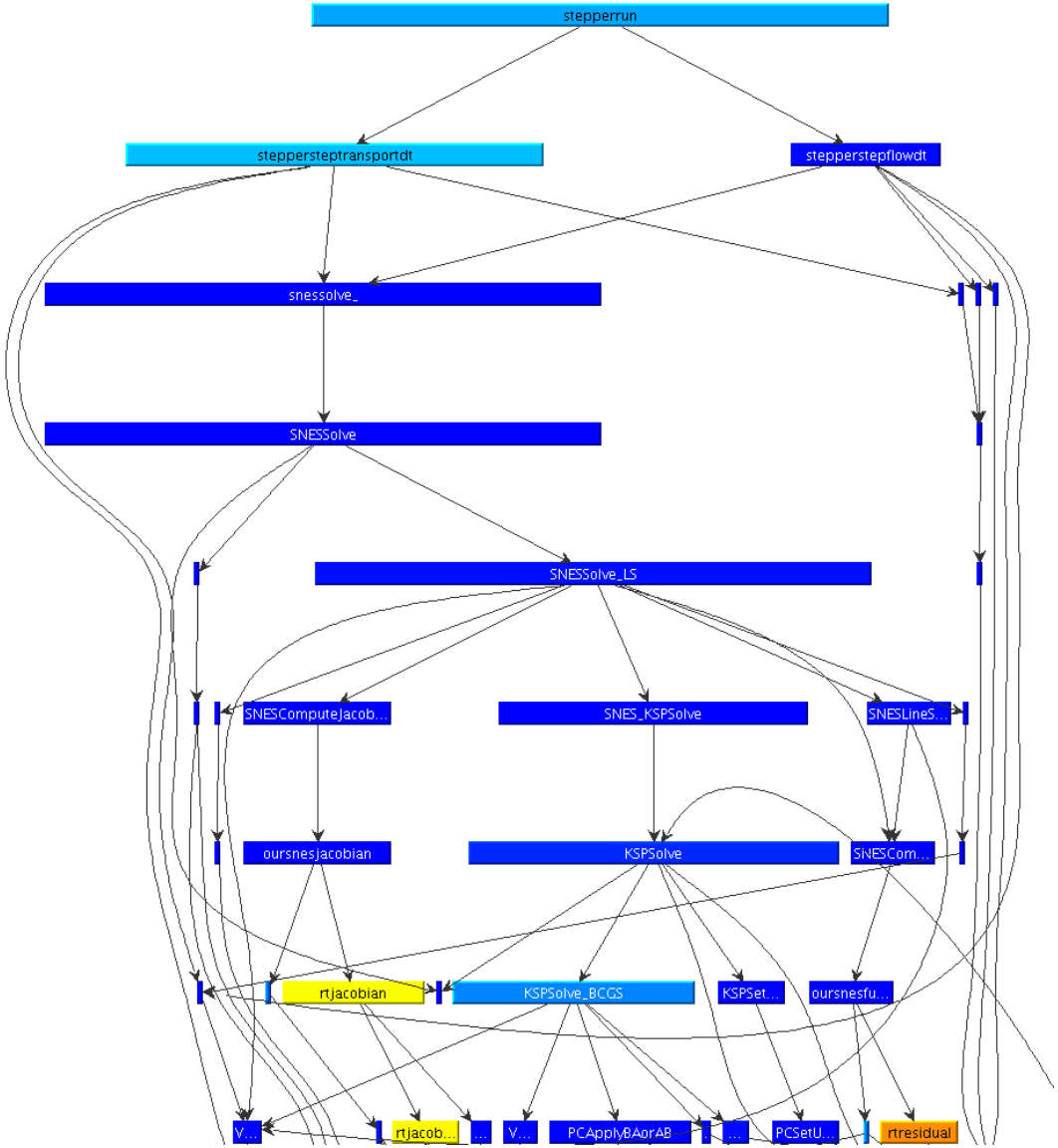
- Automatic application instrumentation
  - both PFLOTRAN application (Fortran) & PETSc library (C)
  - USR routines instrumented by IBM XL & Cray/PGI compilers
  - MPI routines via interposition of instrumented library (PMPI)
- Initial (small-scale) summary measurements used to define filter files specifying all purely computational routines
  - distinct filters required for IBM XL and PGI compilers
- Summary & trace experiments collected using filters
- Post-processing of analysis reports
  - cut to extract timestep loop; incorporation of 3D application topology
- Analysis report examination in GUI

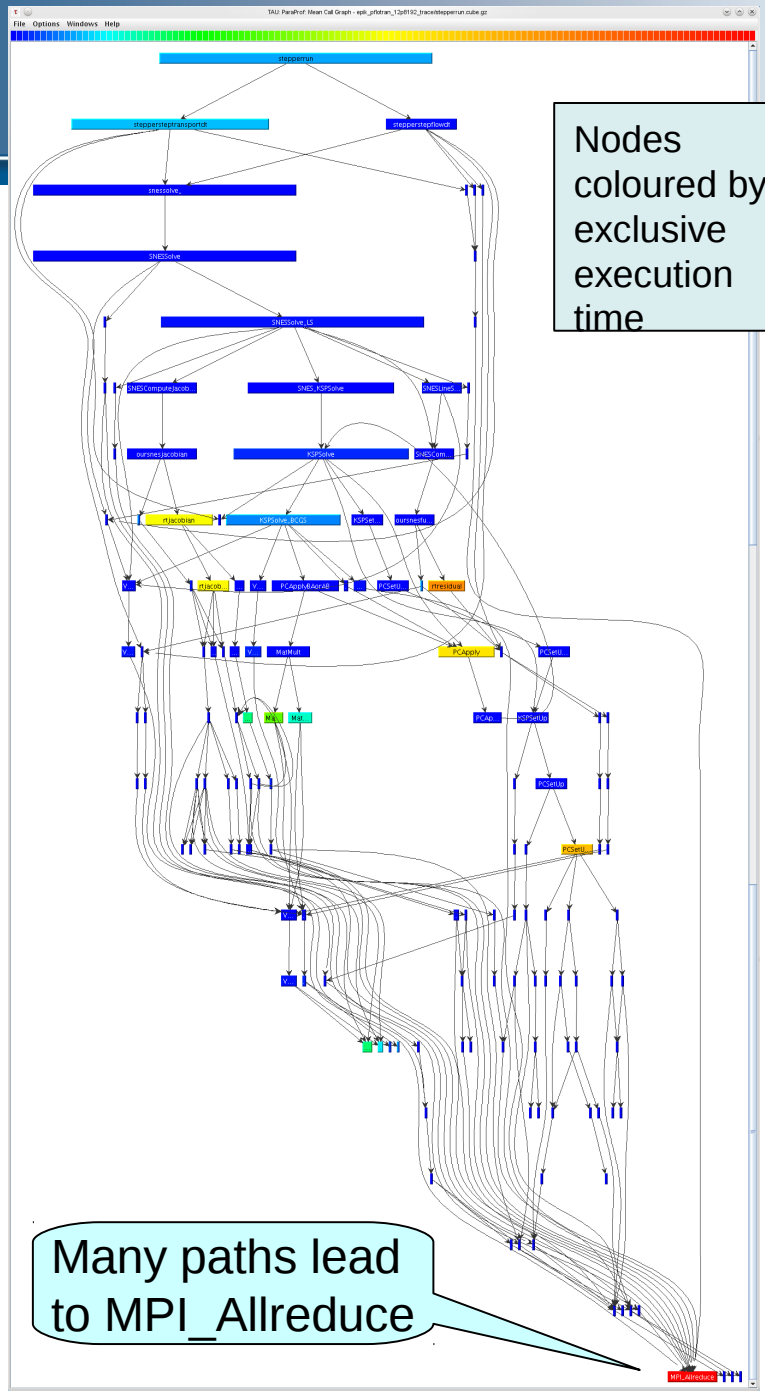
PFLOTRAN structural analysis



- Determined by scoring summary expt using fully-instrumented executable
  - single timestep measurement sufficient
- 1146 PFLOTRAN+PETSc routines executed
  - plus 29 MPI library routines
  - 856 not on a callpath to MPI, purely local calculation (USR)
  - 291 on callpaths to MPI, mixed calculation & comm. (COM)
- Using measurement filter listing all USR routines
  - maximum callpath depth 22 frames
  - 1732 unique callpaths (399 in FLOW, 375 in TRAN)
  - 633 MPI callpaths (121 in FLOW, 114 in TRAN)
- Of 399 FLOW callpaths, 40 missing from TRAN, 20 only in TRAN, 14 with “similar” names
  - richardsjacobian vs rtjacobian, _MPIAIJ vs _MPIBAIJ, etc.

ParaProf view of stepperrun callgraph

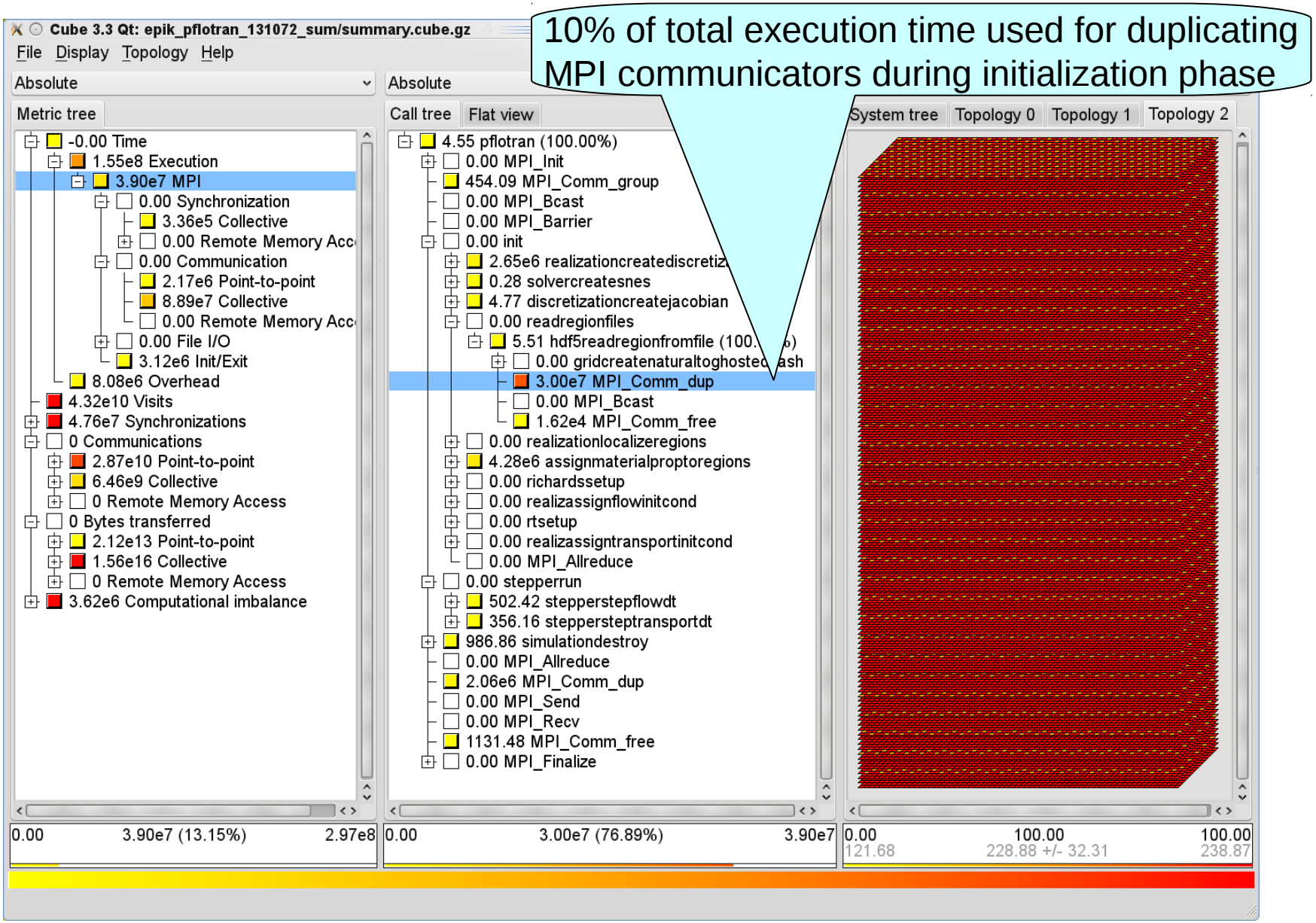




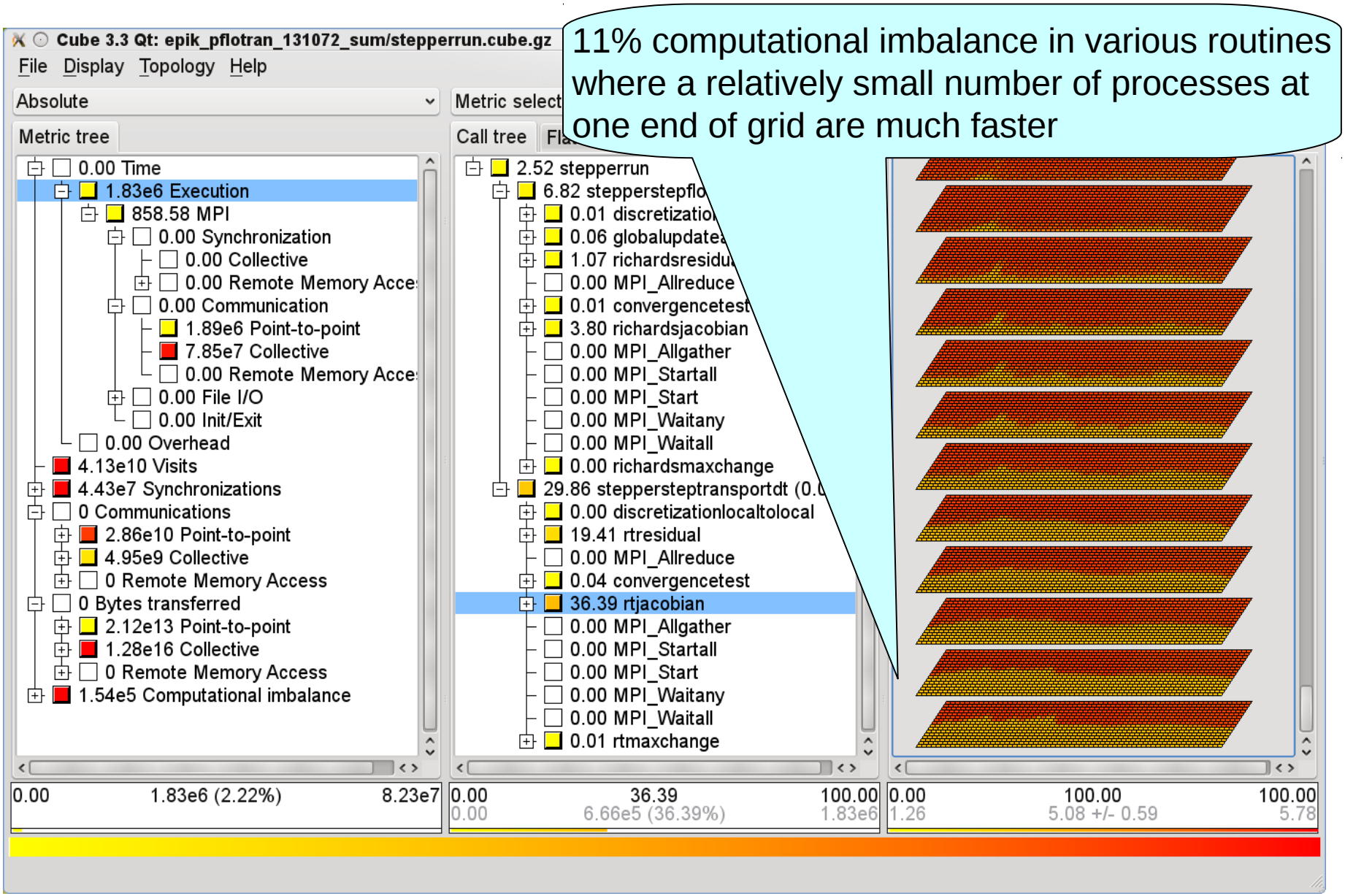
Nodes coloured by exclusive execution time

Many paths lead to MPI_Allreduce

Scalasca summary analysis: entire program

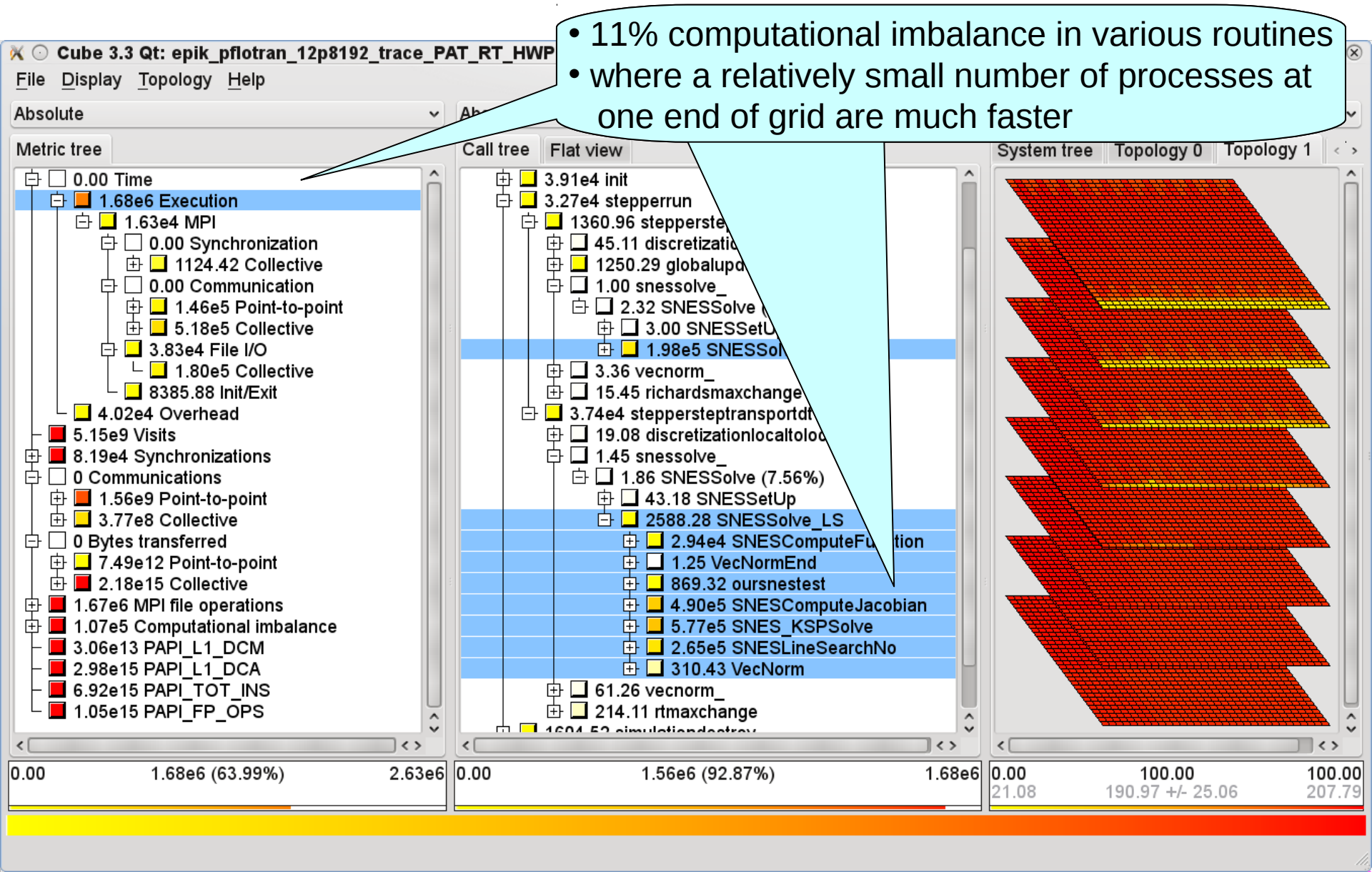


Scalasca summary analysis: stepperrun

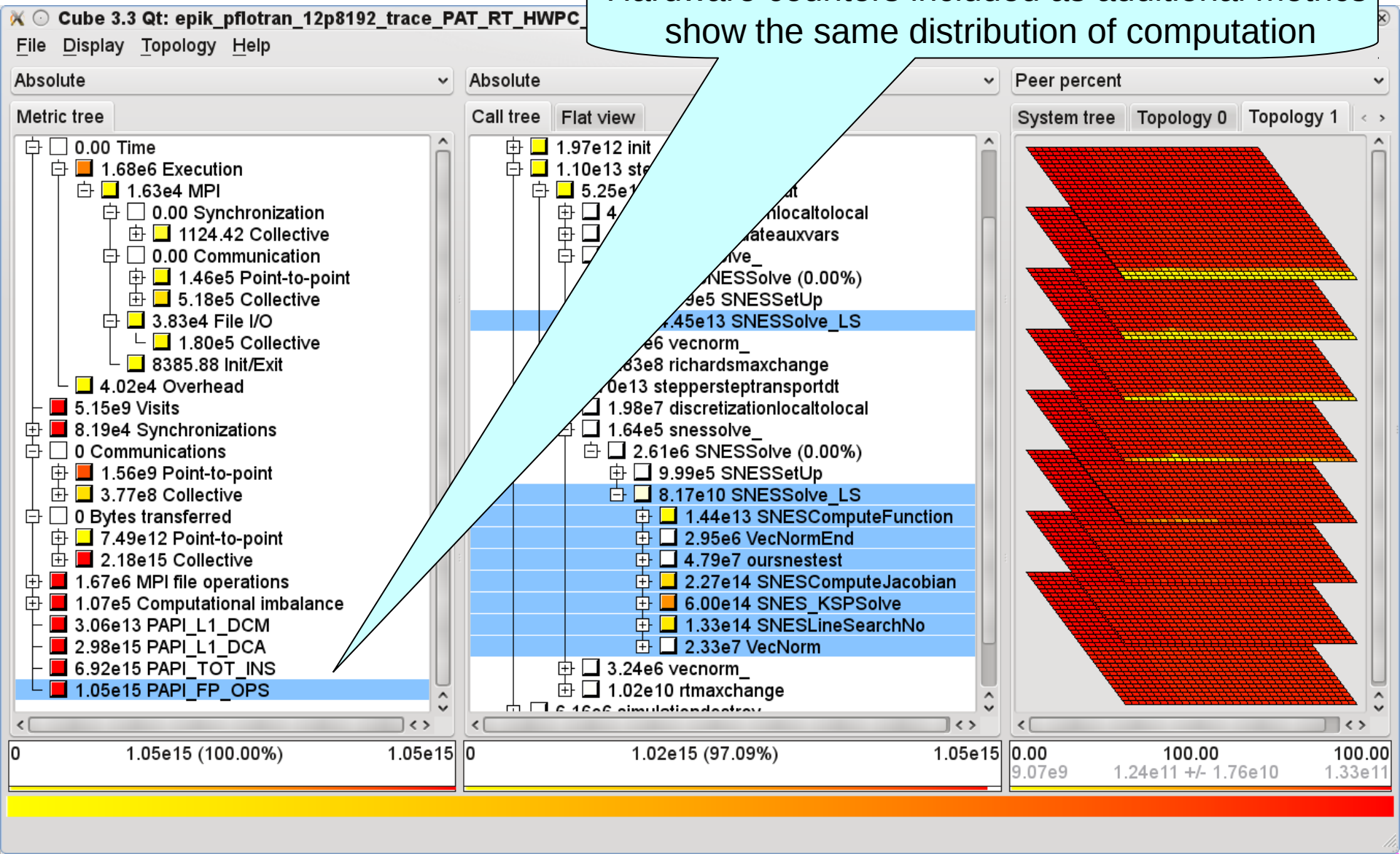


11% computational imbalance in various routines where a relatively small number of processes at one end of grid are much faster

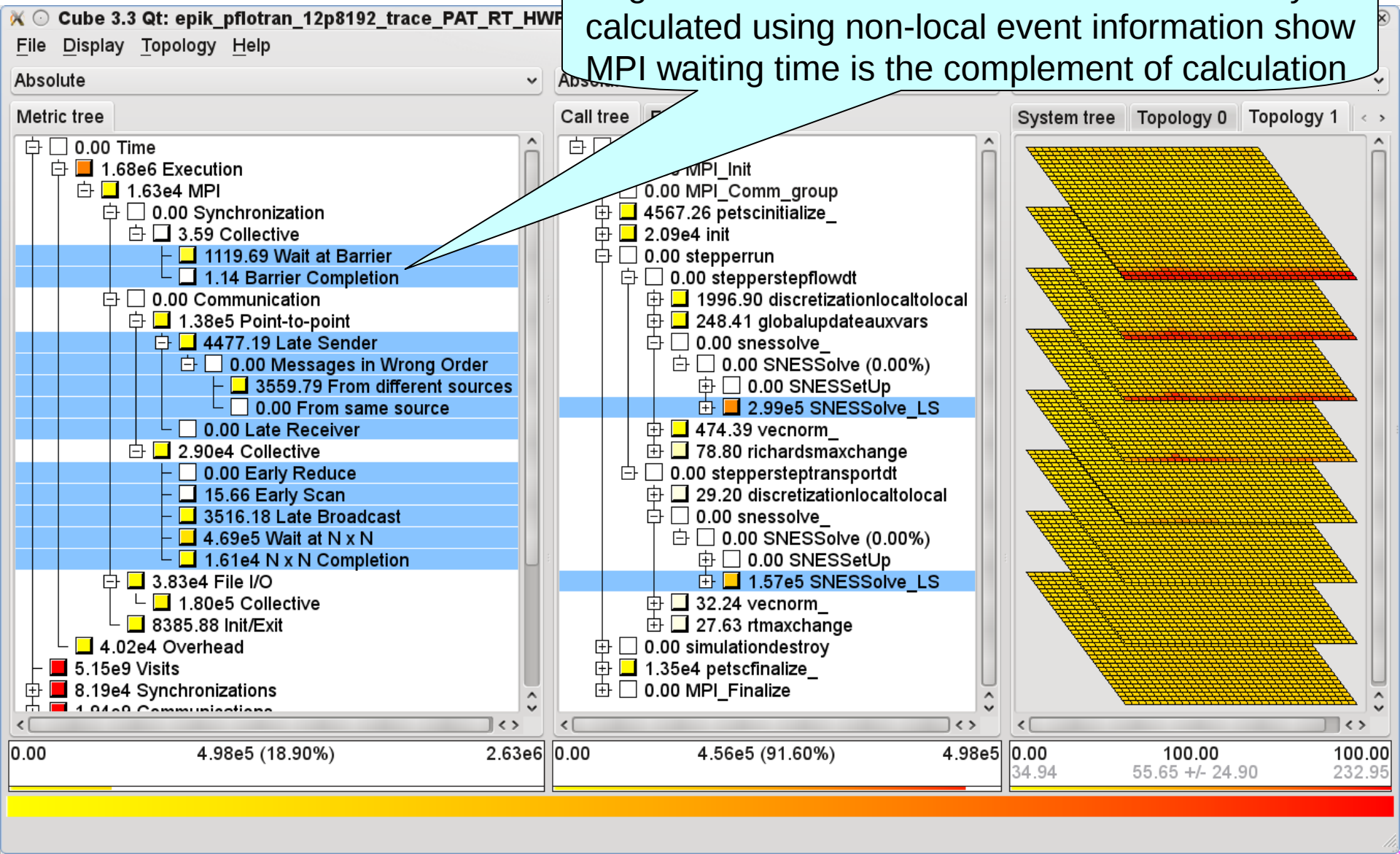
Scalasca analysis: excl. Execution time



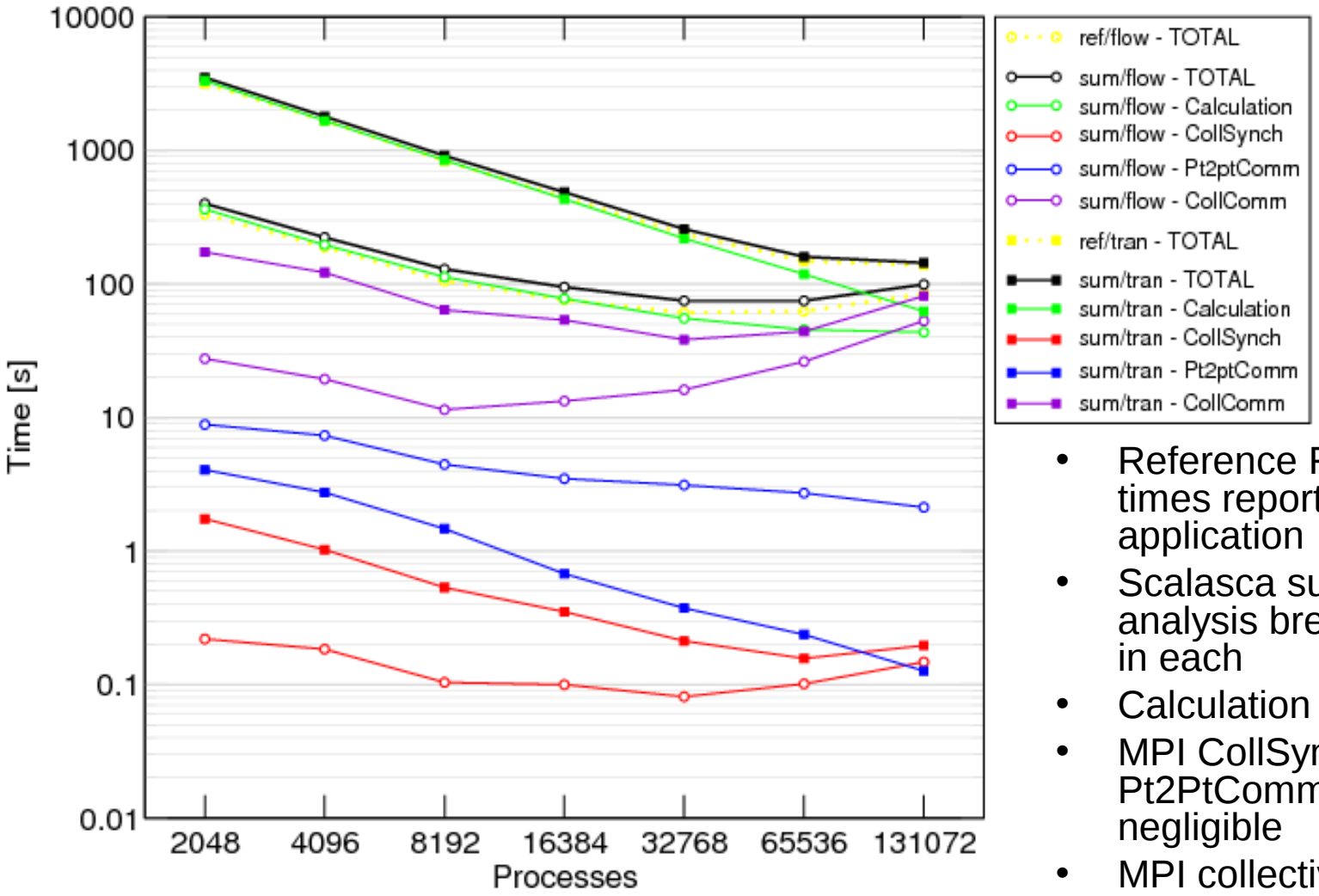
Scalasca analysis: floating-point operations



Scalasca trace analysis: MPI waiting time

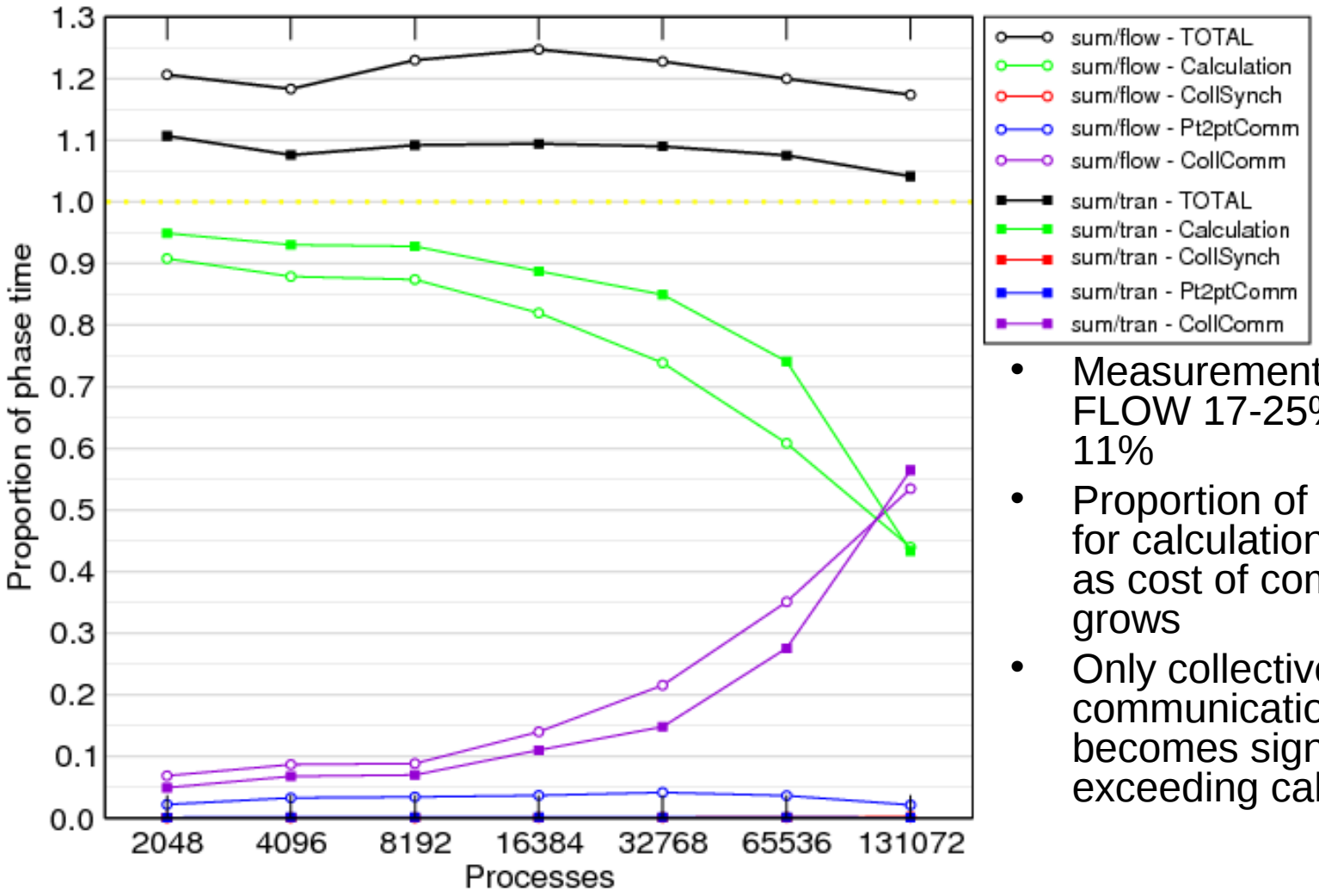


Scalasca scalability analysis



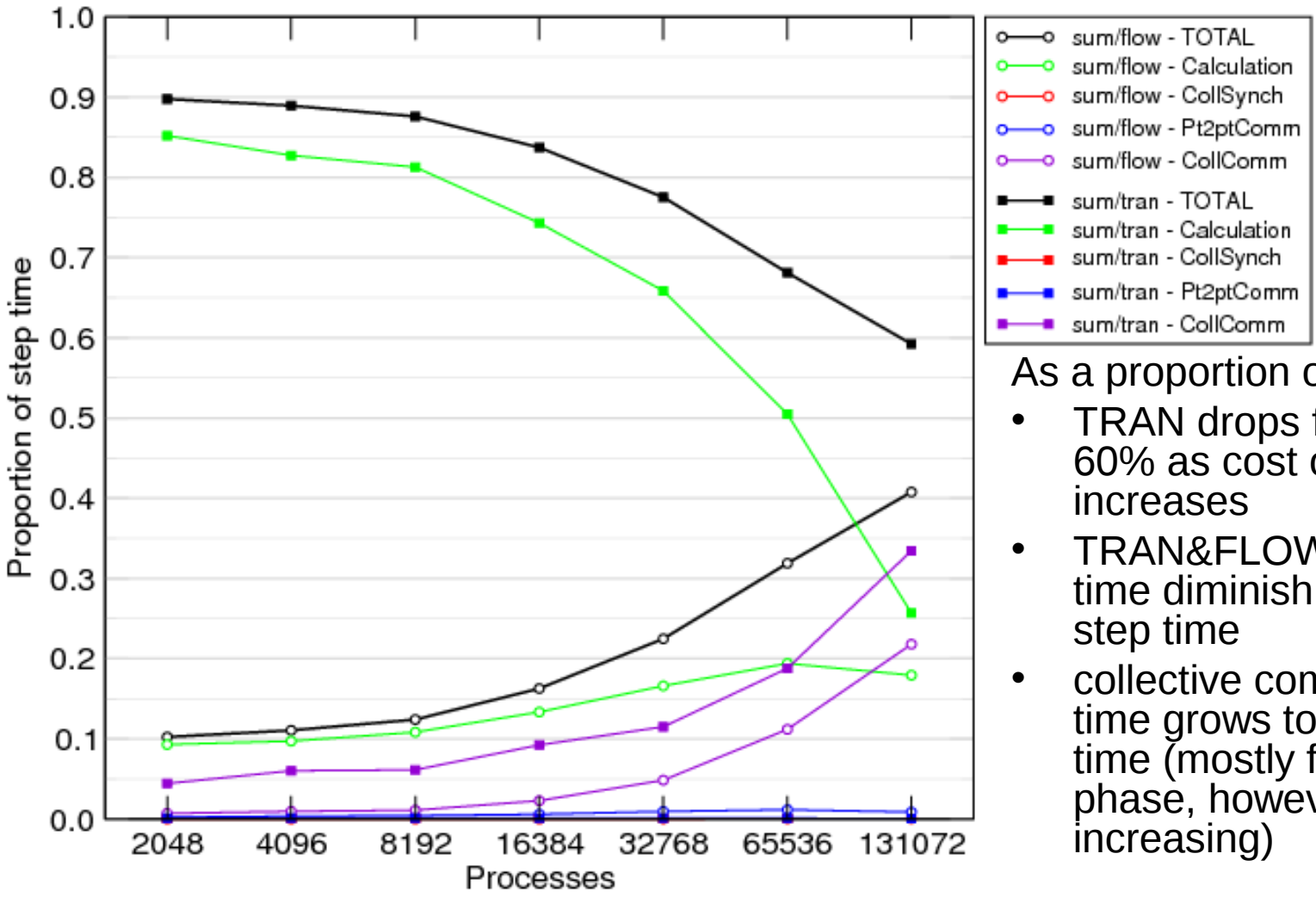
- Reference FLOW&TRAN times reported by application
- Scalasca summary analysis breakdown of time in each
- Calculation scales well
- MPI CollSynch & Pt2PtComm times are negligible
- MPI collective communication becomes a bottleneck

Scalasca scalability analysis



- Measurement dilation: FLOW 17-25% TRAN 4-11%
- Proportion of phase time for calculation diminishes, as cost of communication grows
- Only collective communication cost becomes significant, exceeding calculation times

Scalasca scalability analysis



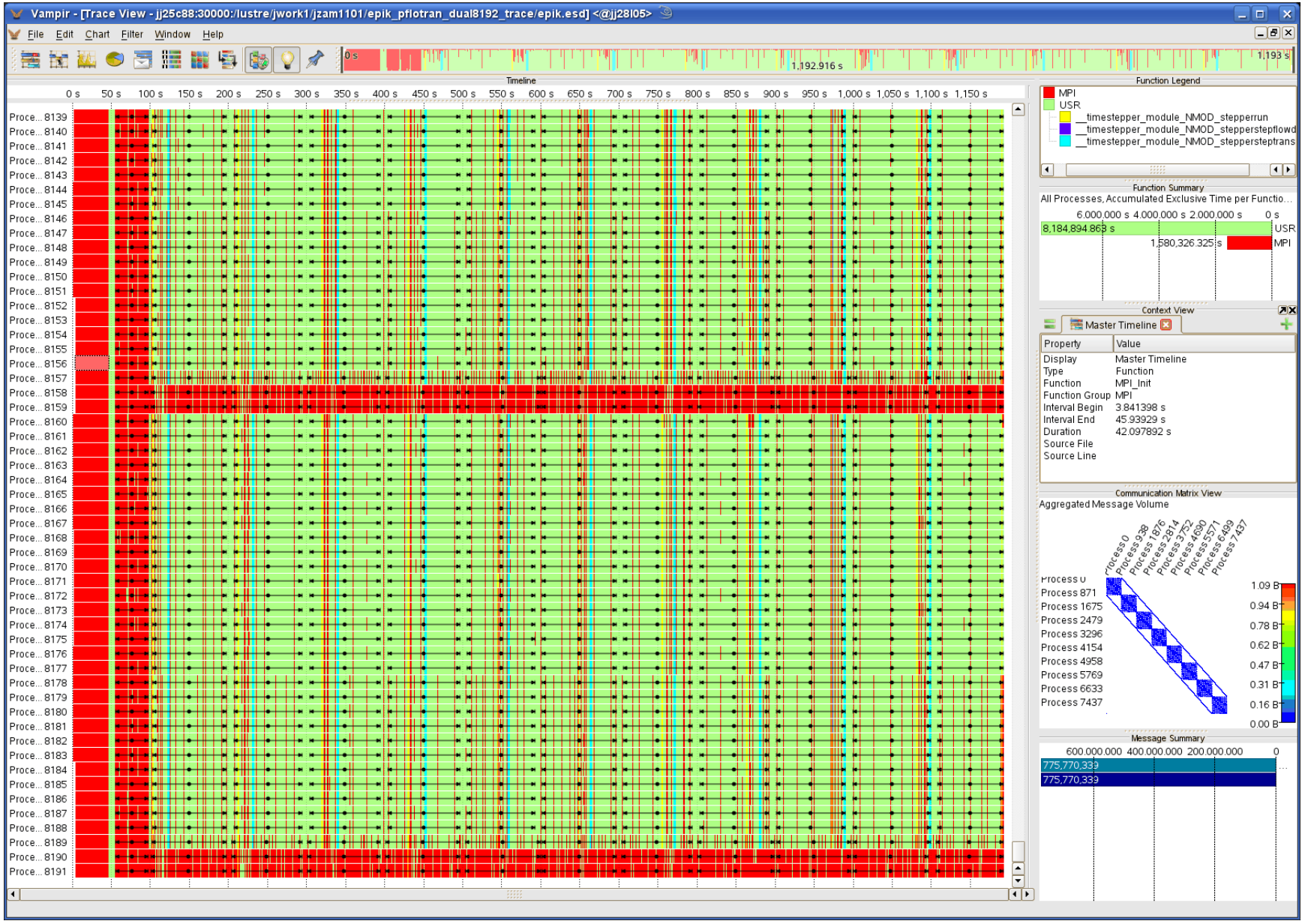
As a proportion of step time

- TRAN drops from 90% to 60% as cost of flow increases
- TRAN&FLOW calculation time diminish to 44% of step time
- collective communication time grows to 55% of step time (mostly from TRAN phase, however, FLOW increasing)

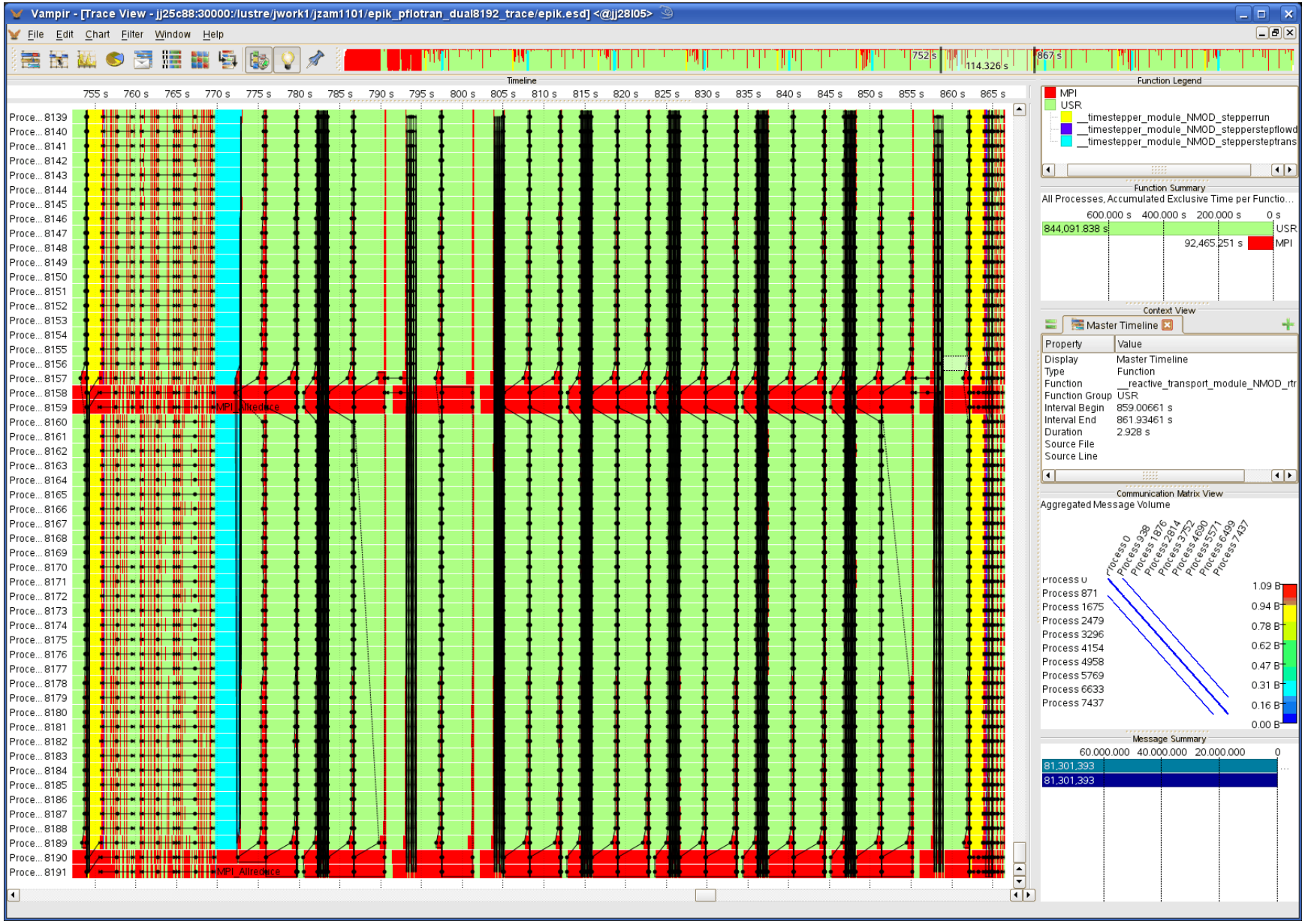
Usage of Vampir

- Collect traces with up to 16,384 MPI processes on BG/P for parallel analysis with Vampir7
- Use of manual instrumentation API for selective tracing
  - disable measurement during initialization phase and only enable trace of one timestep
  - reduces traces to a managable size
- Interactive exploratory analysis of traces
  - cluster similar processes
  - zoom into time interval of interest
  - scroll/zoom timelines for processes
  - investigate communication patterns
  - examine message distributions

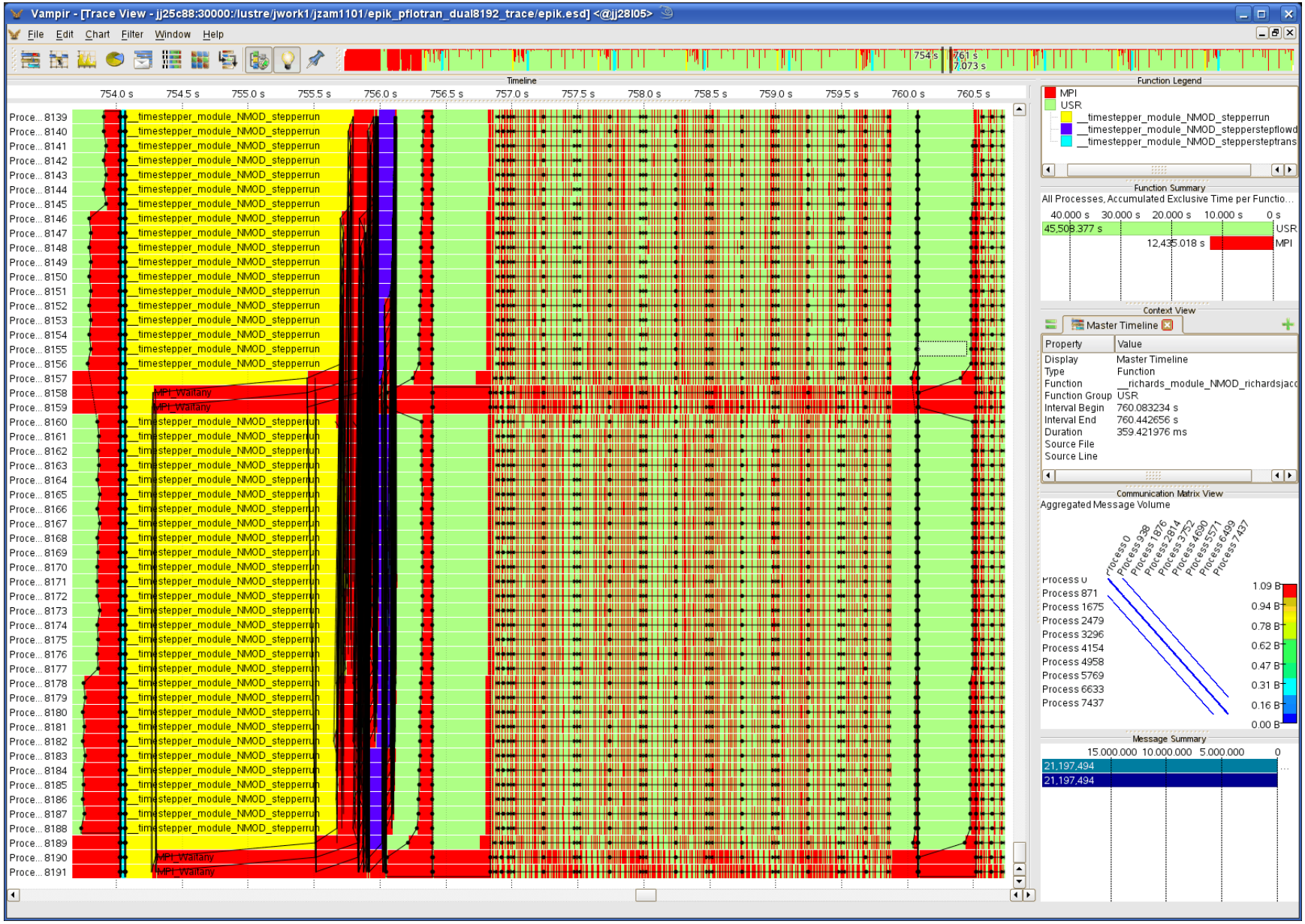
Full execution time-line



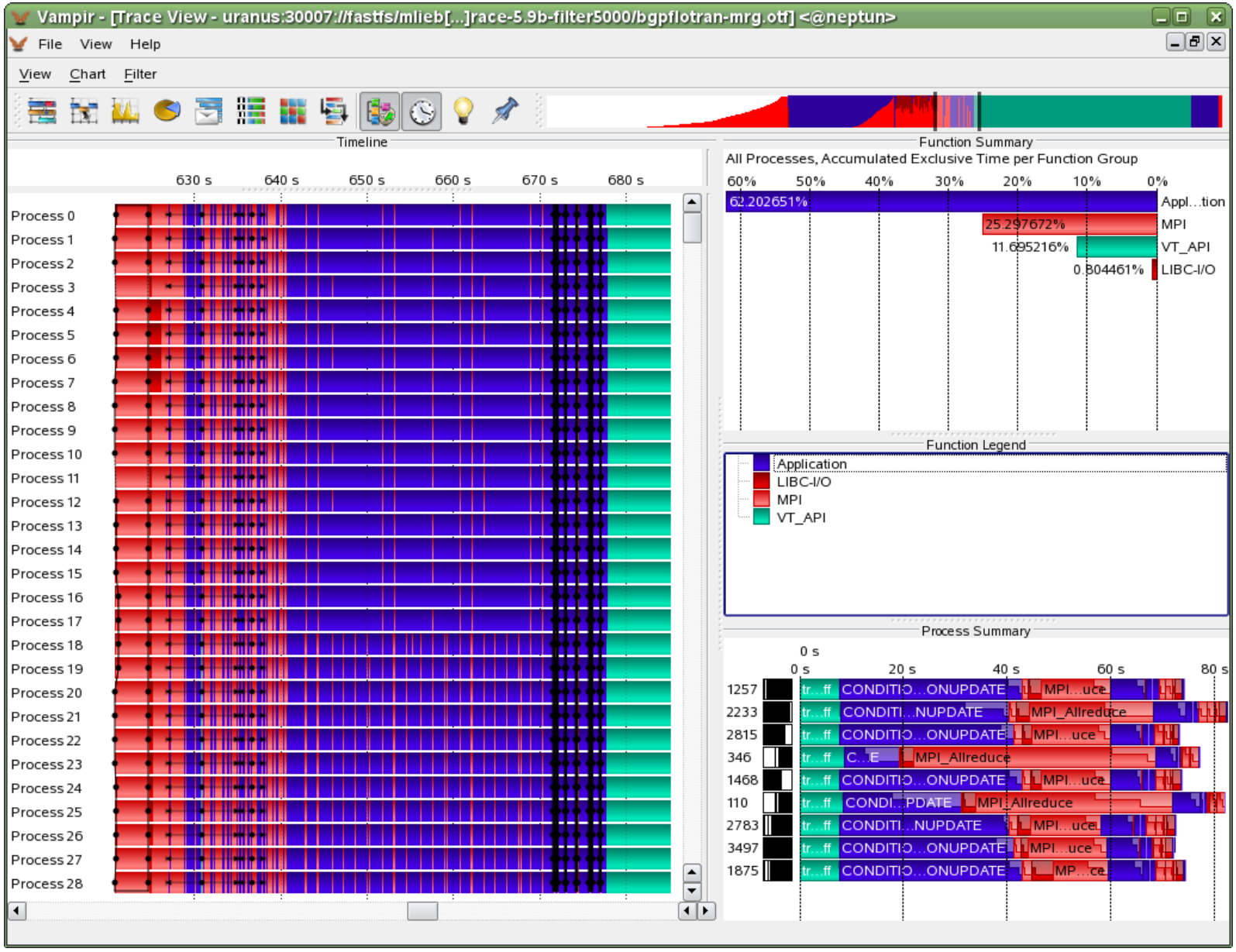
Timestep 7 zoom



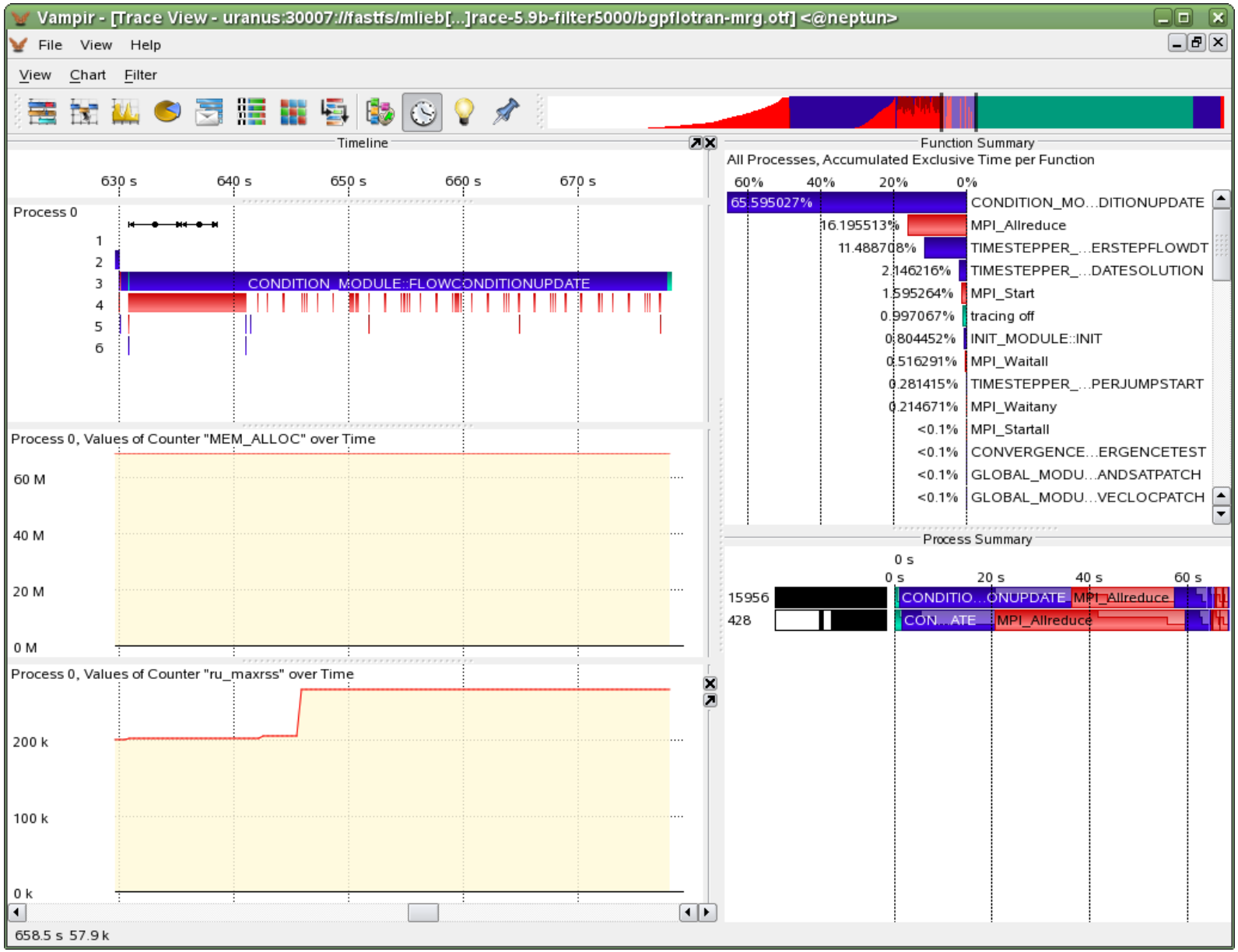
PFLOW phase zoom



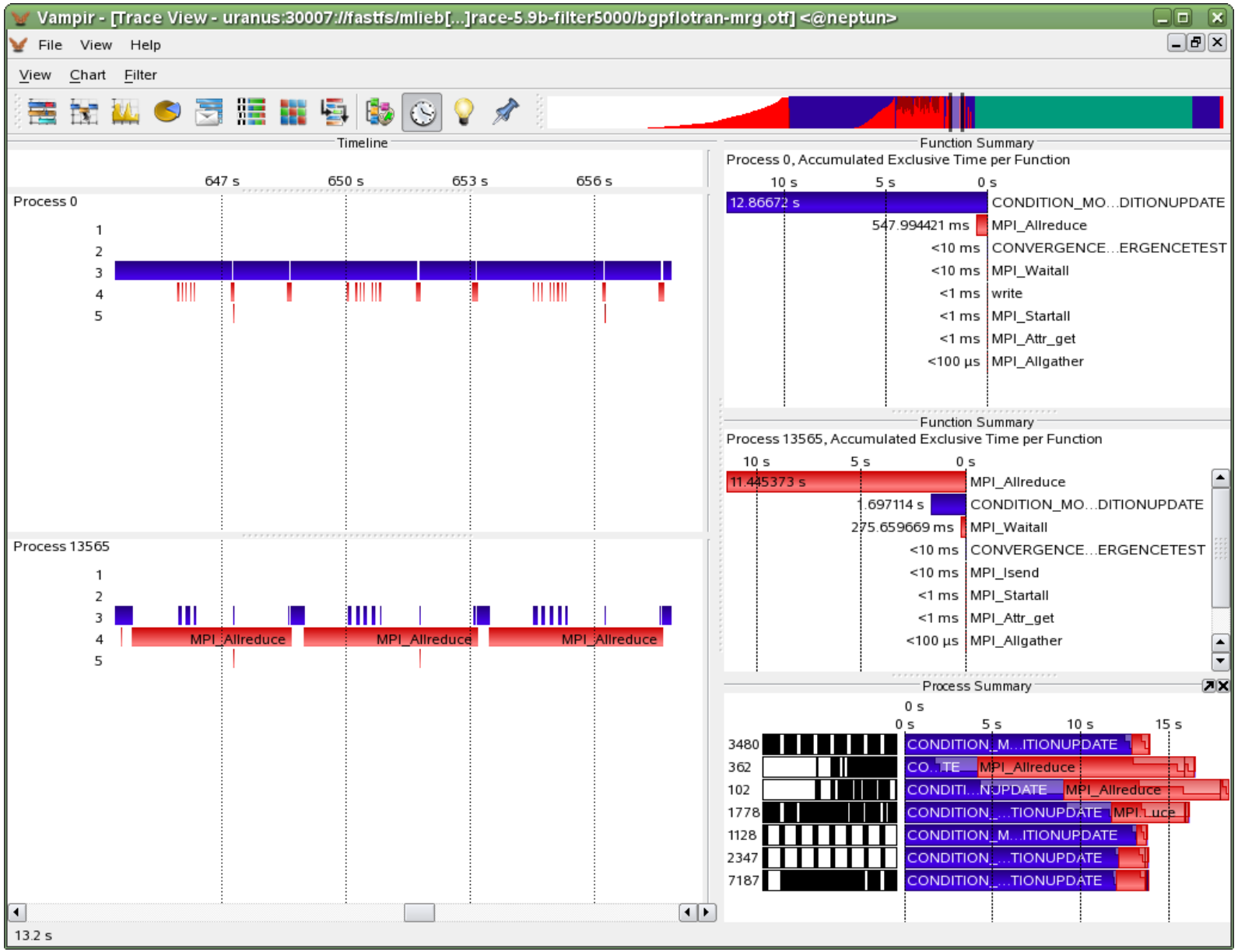
End of init phase + 1st timestep



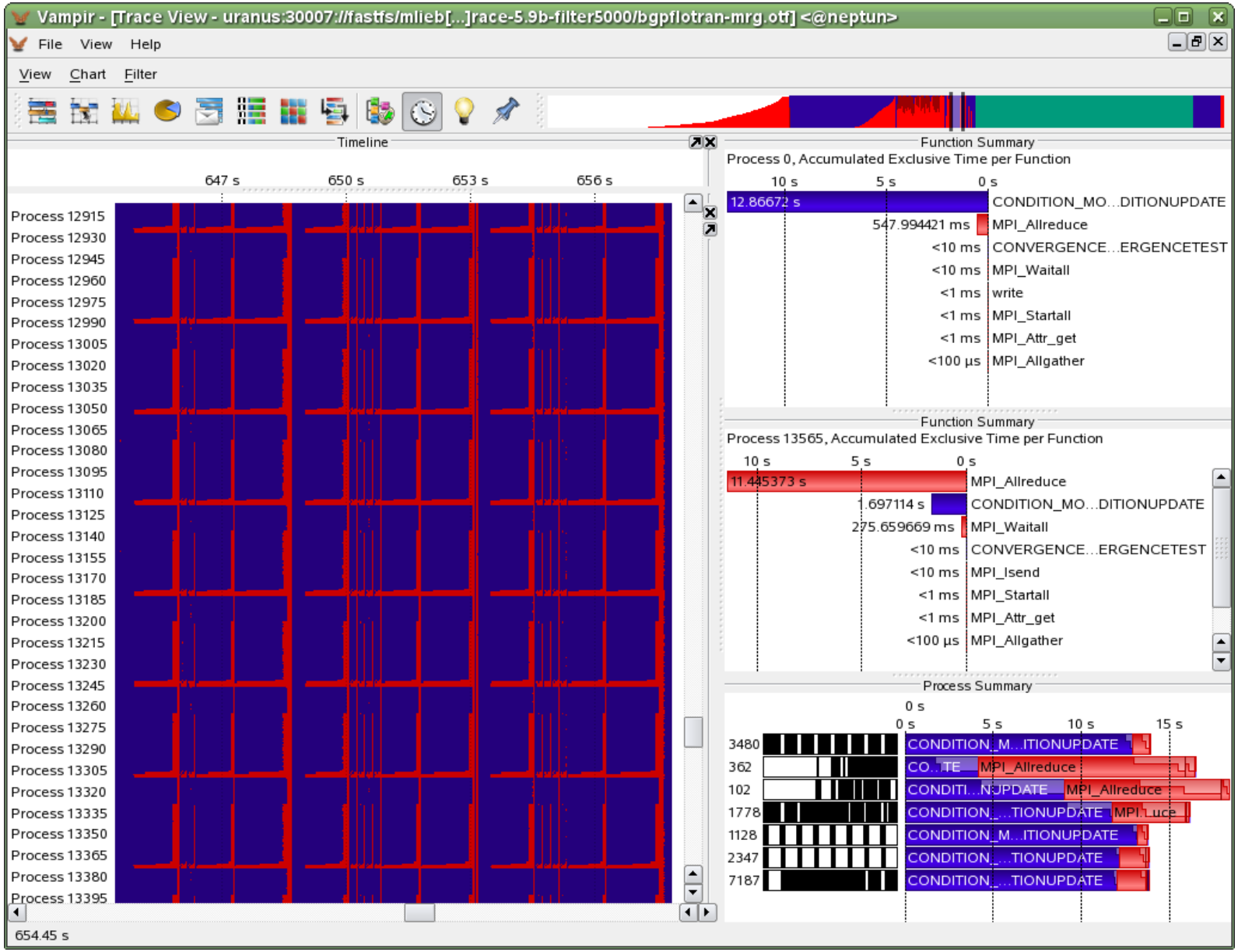
Clustering of processes



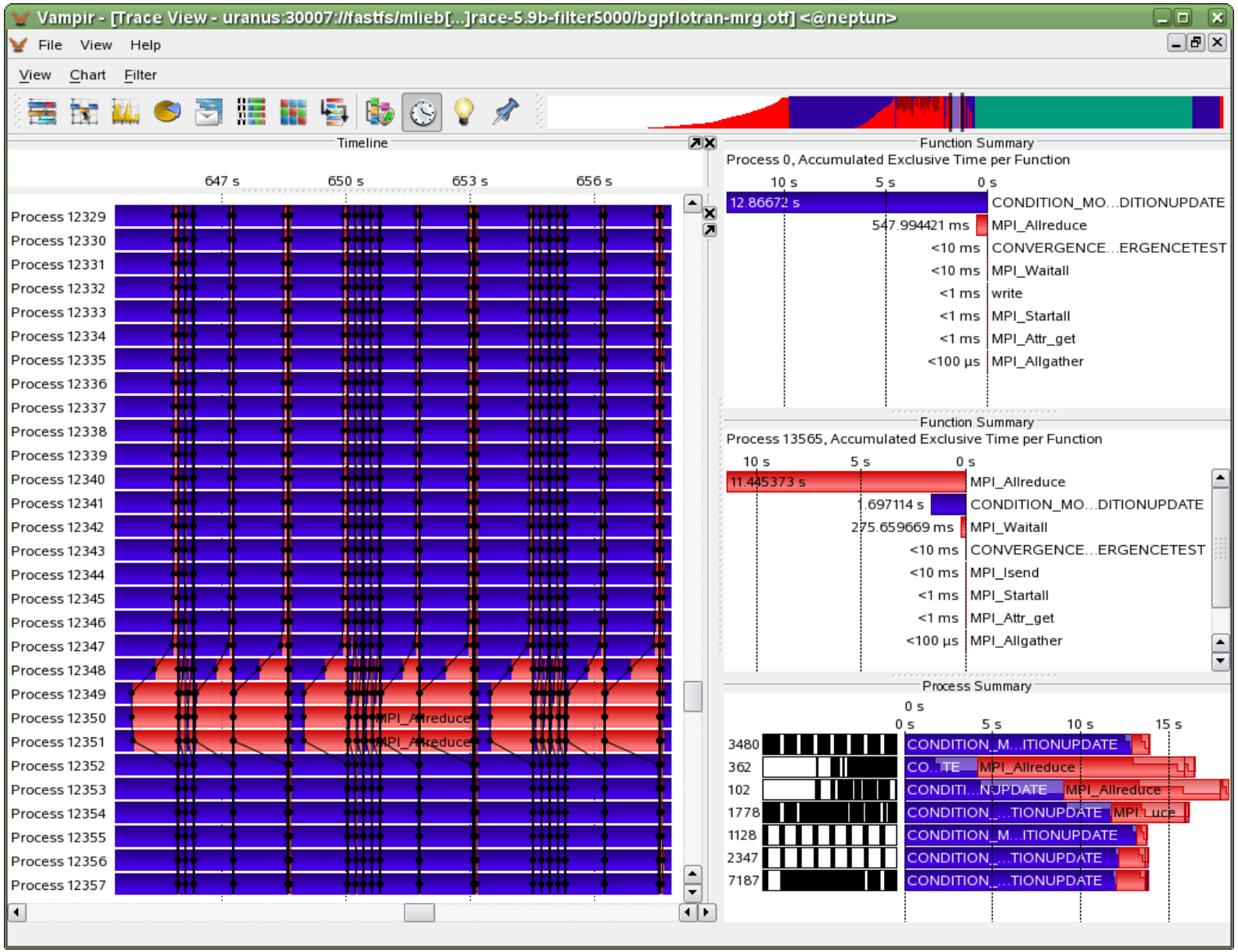
Comparing process timelines



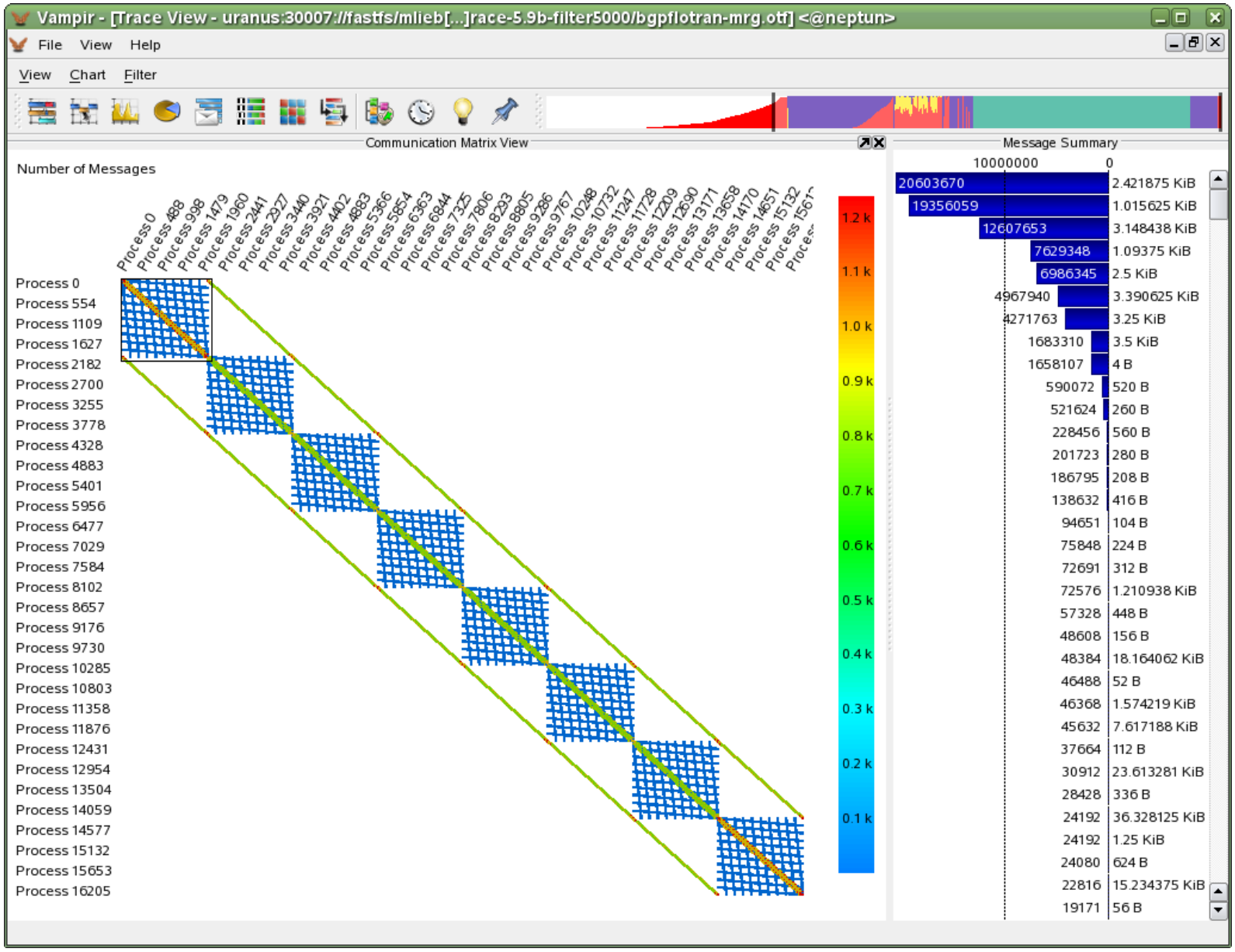
1st timestep: imbalance



1st timestep: imbalance (zoom)



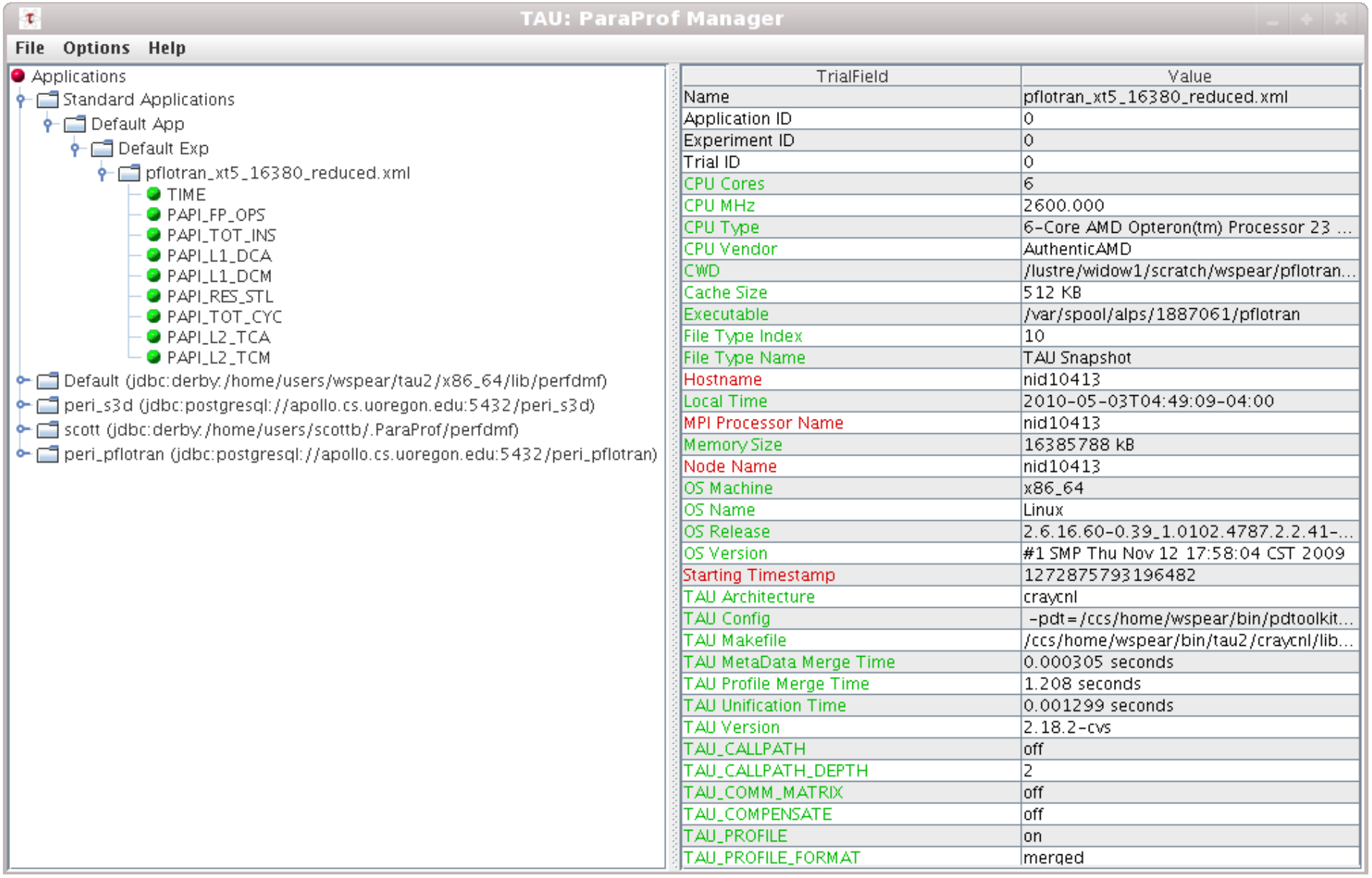
Comm matrix + message hist



Usage of TAU

- Use PDT for selective instrumentation
  - Select user source routines taking more than 1% of exclusive execution time
    - 4 from PFLOTRAN, 19 from PETSc [+44 MPI]
- Collect profiles on Jaguar Cray XT5 from runs with varying numbers of MPI processes
  - Include PAPI preset hardware counters
- Use ParaProf Manager to browse experiments
  - Examine 2D & 3D graphical profile, histogram and callgraph presentations

ParaProf Manager (16380)



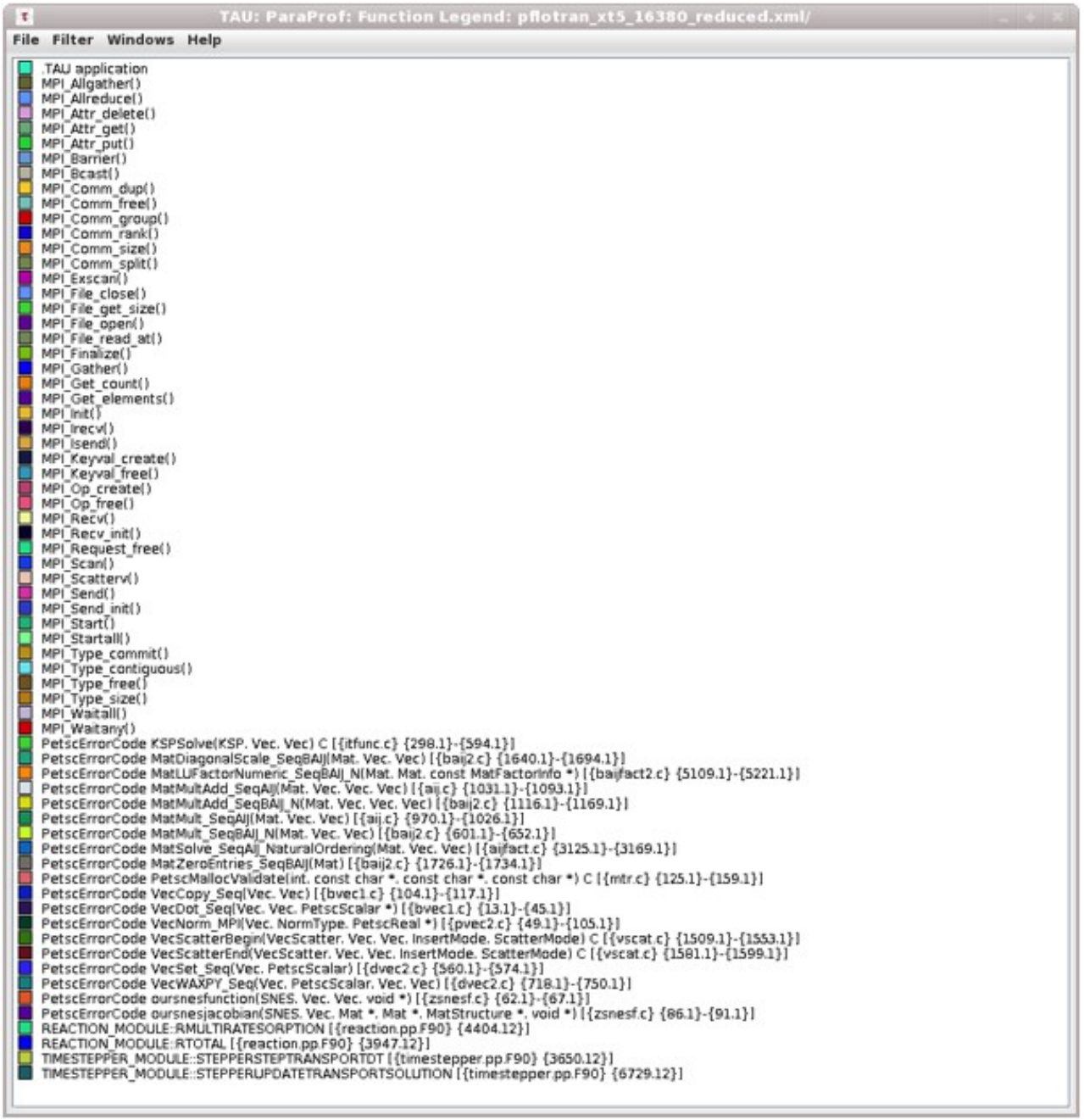
TrialField	Value
Name	pflotran_xt5_16380_reduced.xml
Application ID	0
Experiment ID	0
Trial ID	0
CPU Cores	6
CPU MHz	2600.000
CPU Type	6-Core AMD Opteron(tm) Processor 23 ...
CPU Vendor	AuthenticAMD
CWD	/lustre/widow1/scratch/wspear/pflotran...
Cache Size	512 KB
Executable	/var/spool/alps/1887061/pflotran
File Type Index	10
File Type Name	TAU Snapshot
Hostname	nid10413
Local Time	2010-05-03T04:49:09-04:00
MPI Processor Name	nid10413
Memory Size	16385788 kB
Node Name	nid10413
OS Machine	x86_64
OS Name	Linux
OS Release	2.6.16.60-0.39_1.0102.4787.2.2.41-...
OS Version	#1 SMP Thu Nov 12 17:58:04 CST 2009
Starting Timestamp	1272875793196482
TAU Architecture	craycnl
TAU Config	-pdt=/ccs/home/wspear/bin/pdtoolkit...
TAU Makefile	/ccs/home/wspear/bin/tau2/craycnl/lib...
TAU MetaData Merge Time	0.000305 seconds
TAU Profile Merge Time	1.208 seconds
TAU Unification Time	0.001299 seconds
TAU Version	2.18.2-cvs
TAU_CALLPATH	off
TAU_CALLPATH_DEPTH	2
TAU_COMM_MATRIX	off
TAU_COMPENSATE	off
TAU_PROFILE	on
TAU_PROFILE_FORMAT	merged

.TAU_application

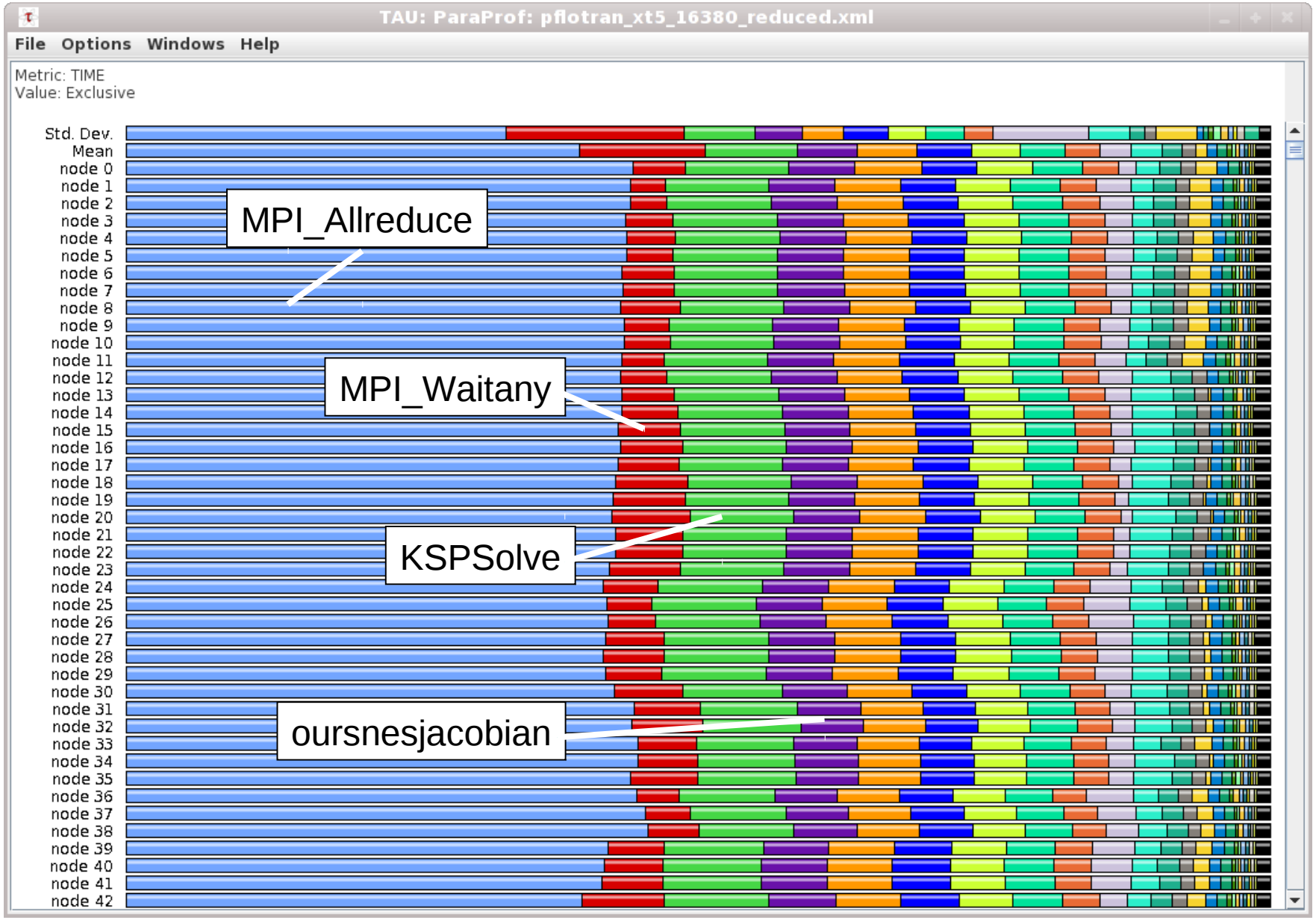
MPI

PETSc

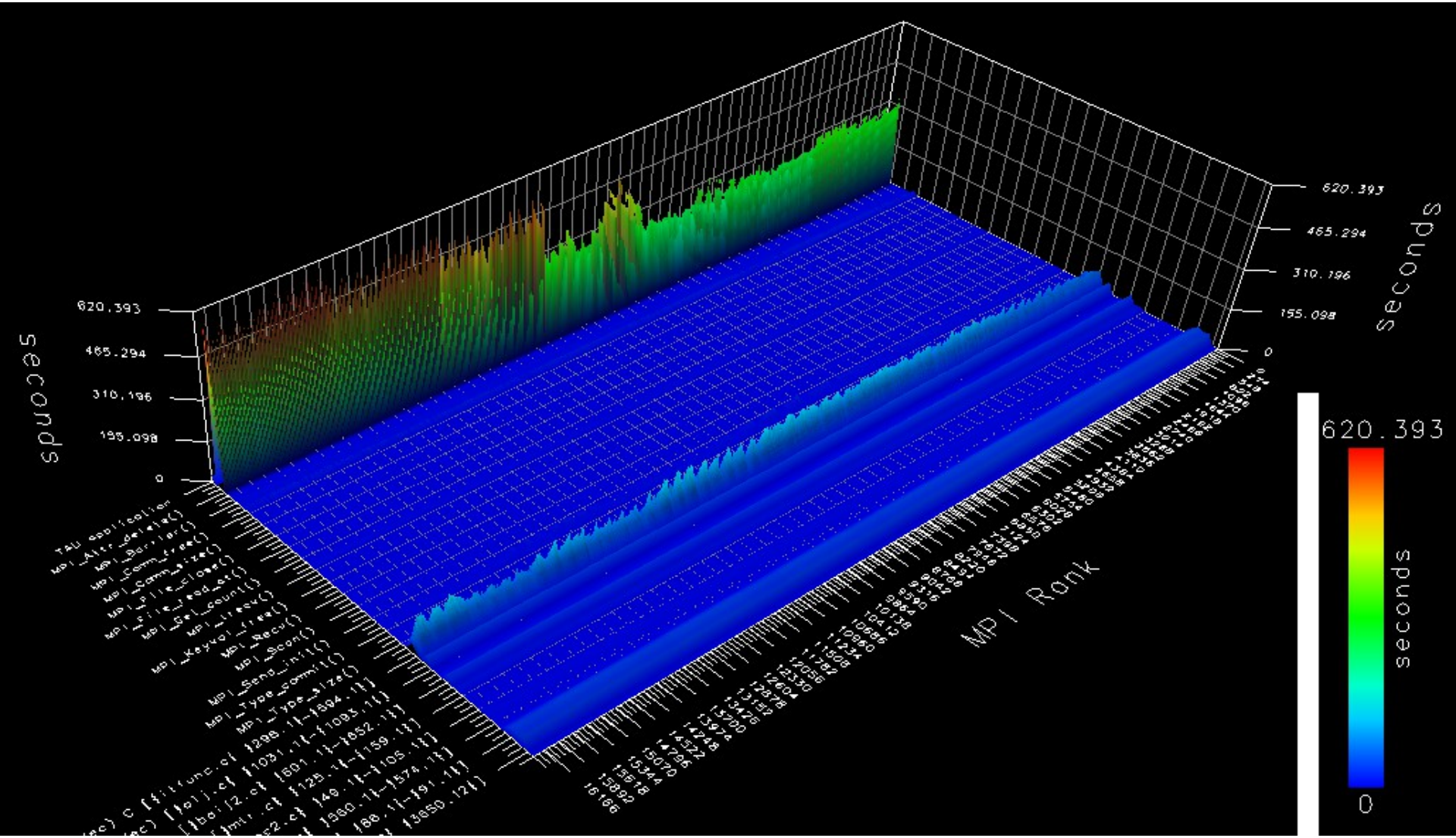
PFLOTRAN



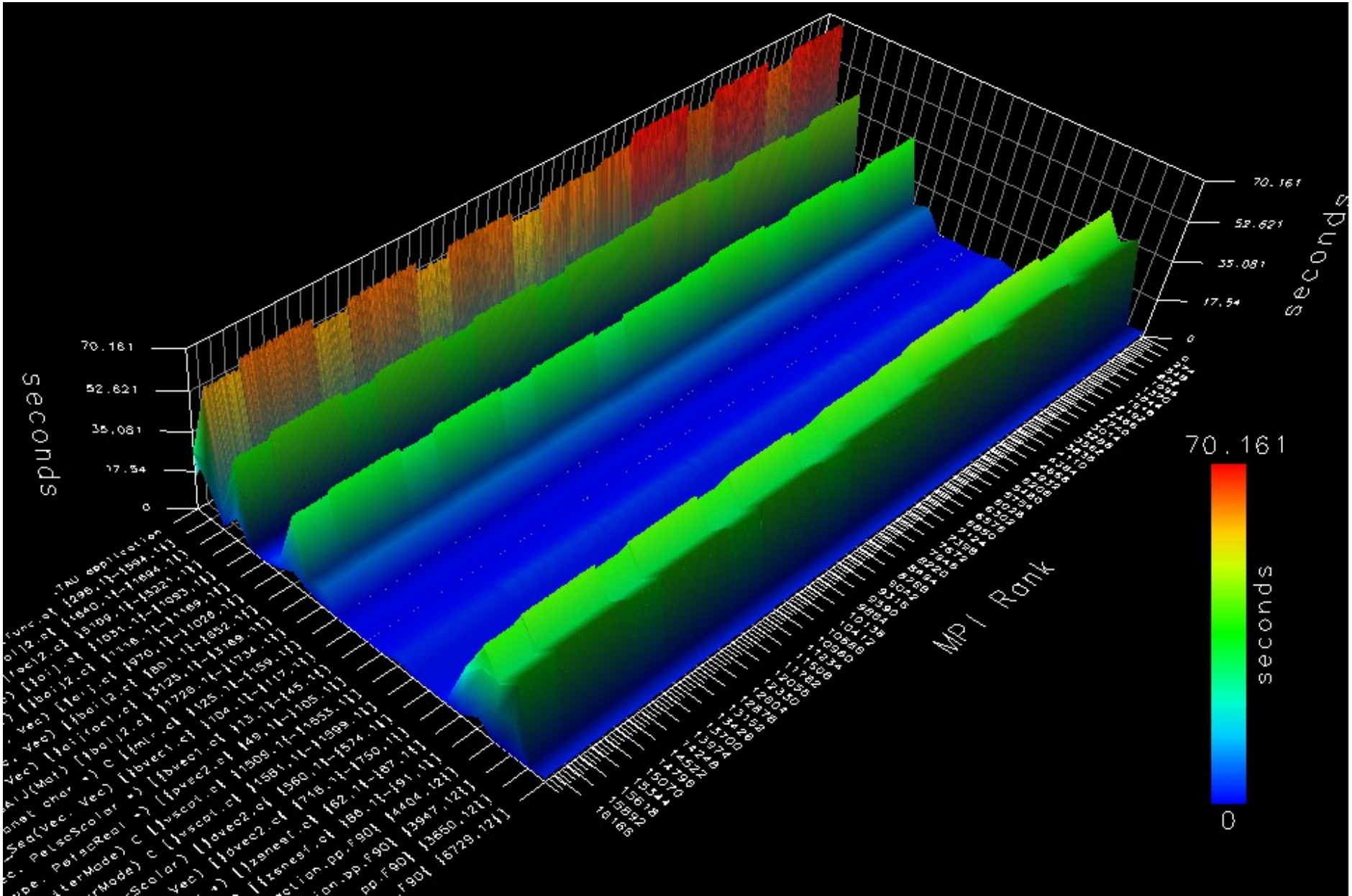
ParaProf full profile (16380)



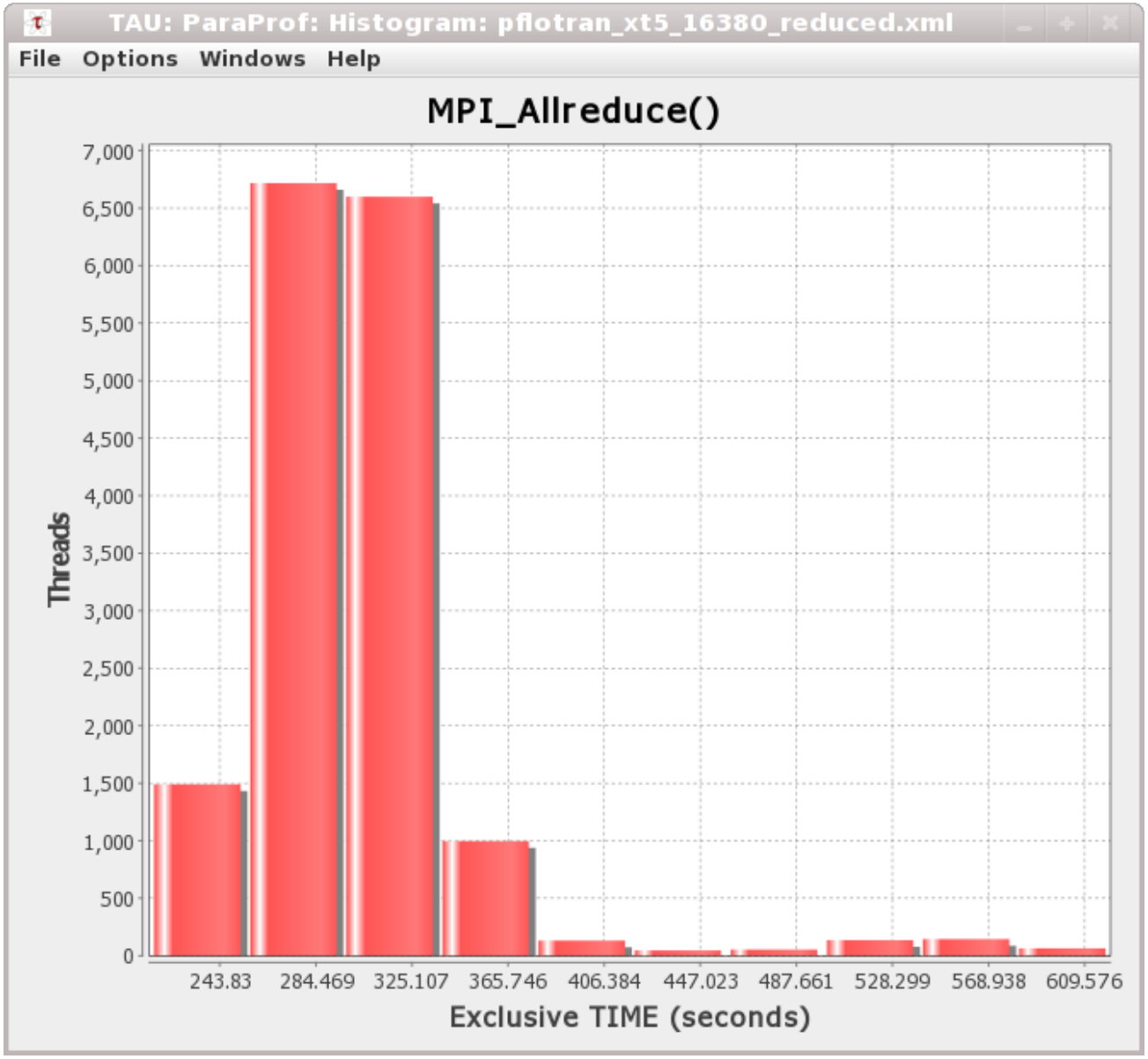
ParaProf 3D full profile (16380)



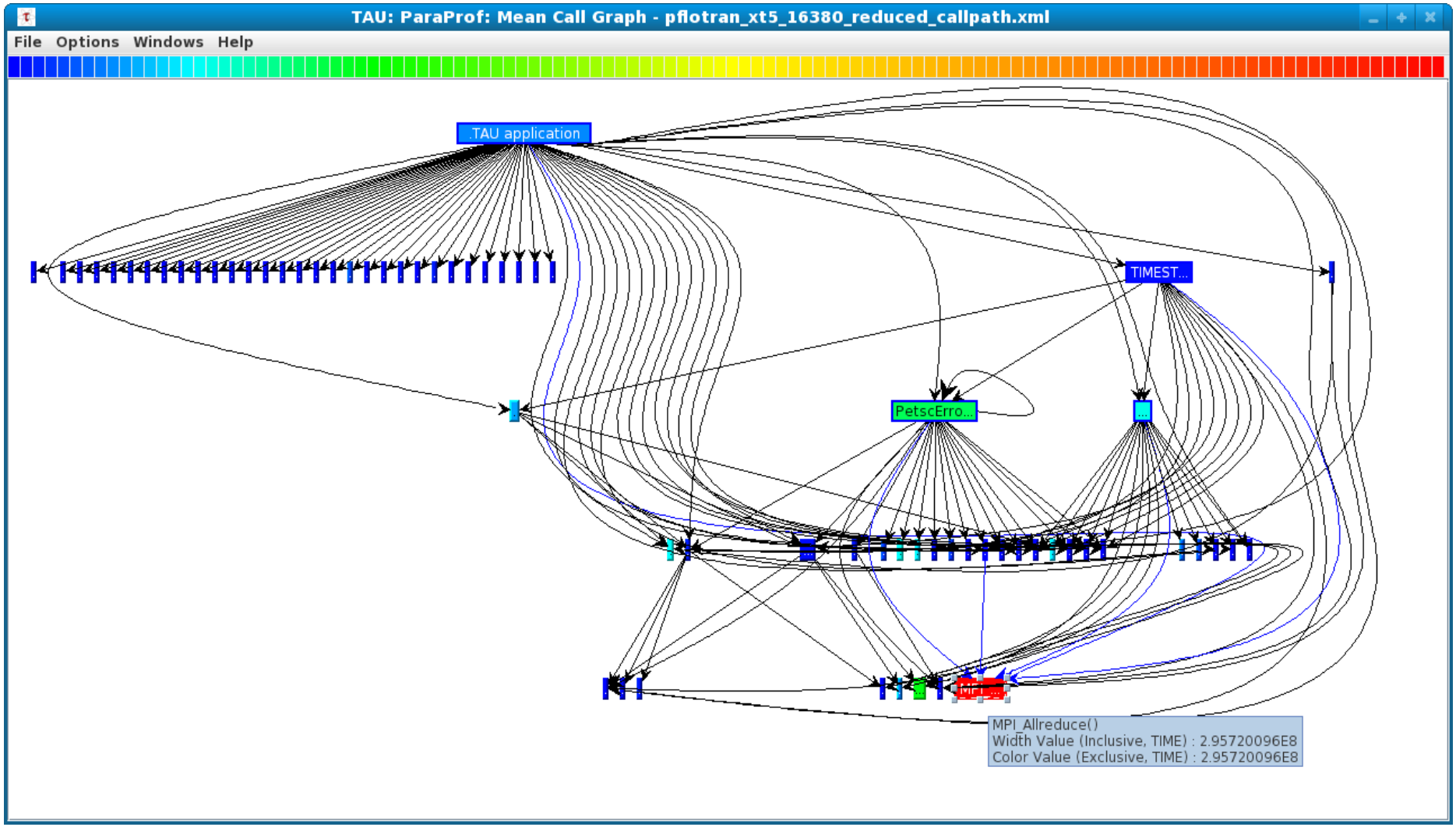
3D profile (w/o MPI_Allreduce)



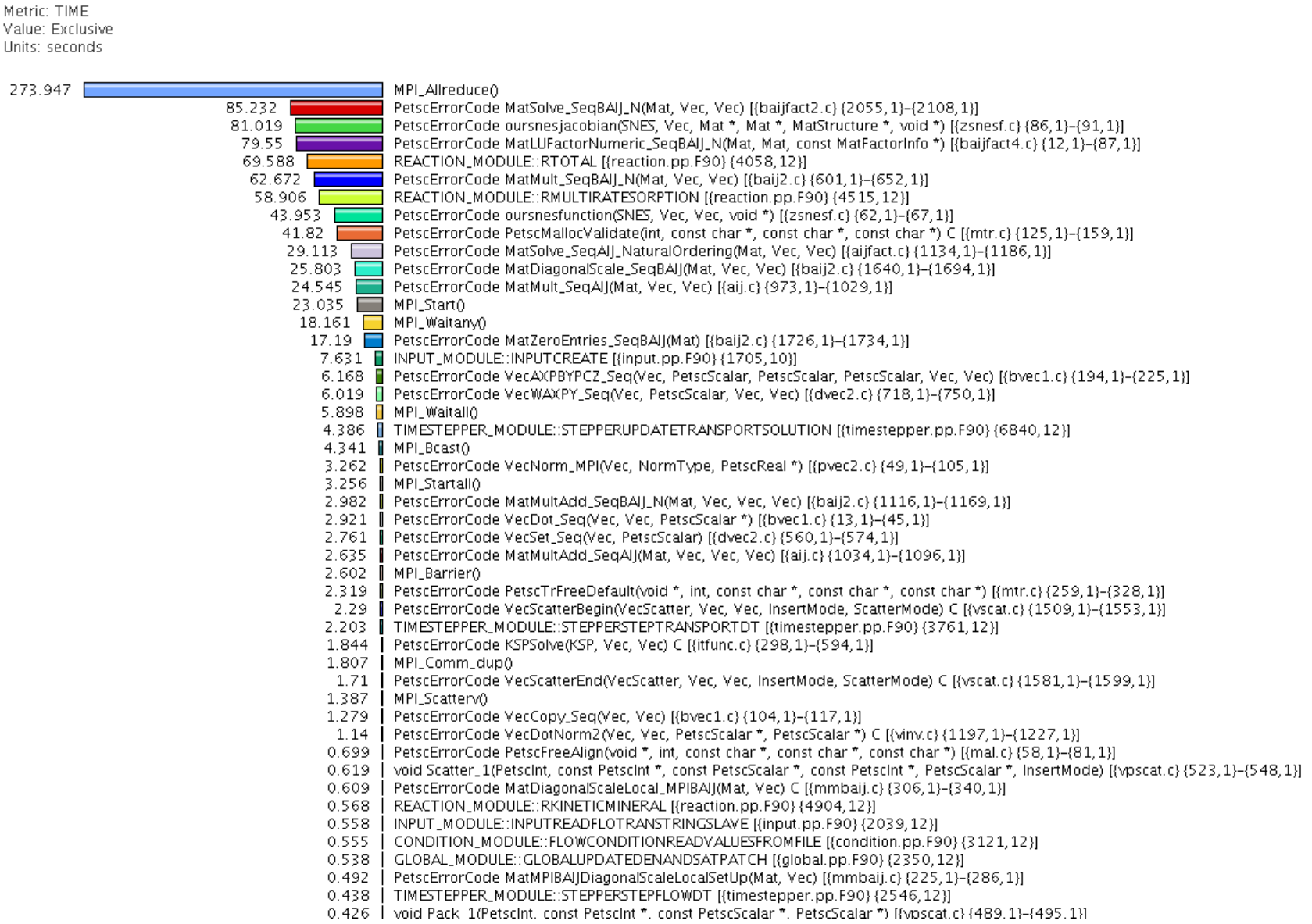
ParaProf histogram (16380)



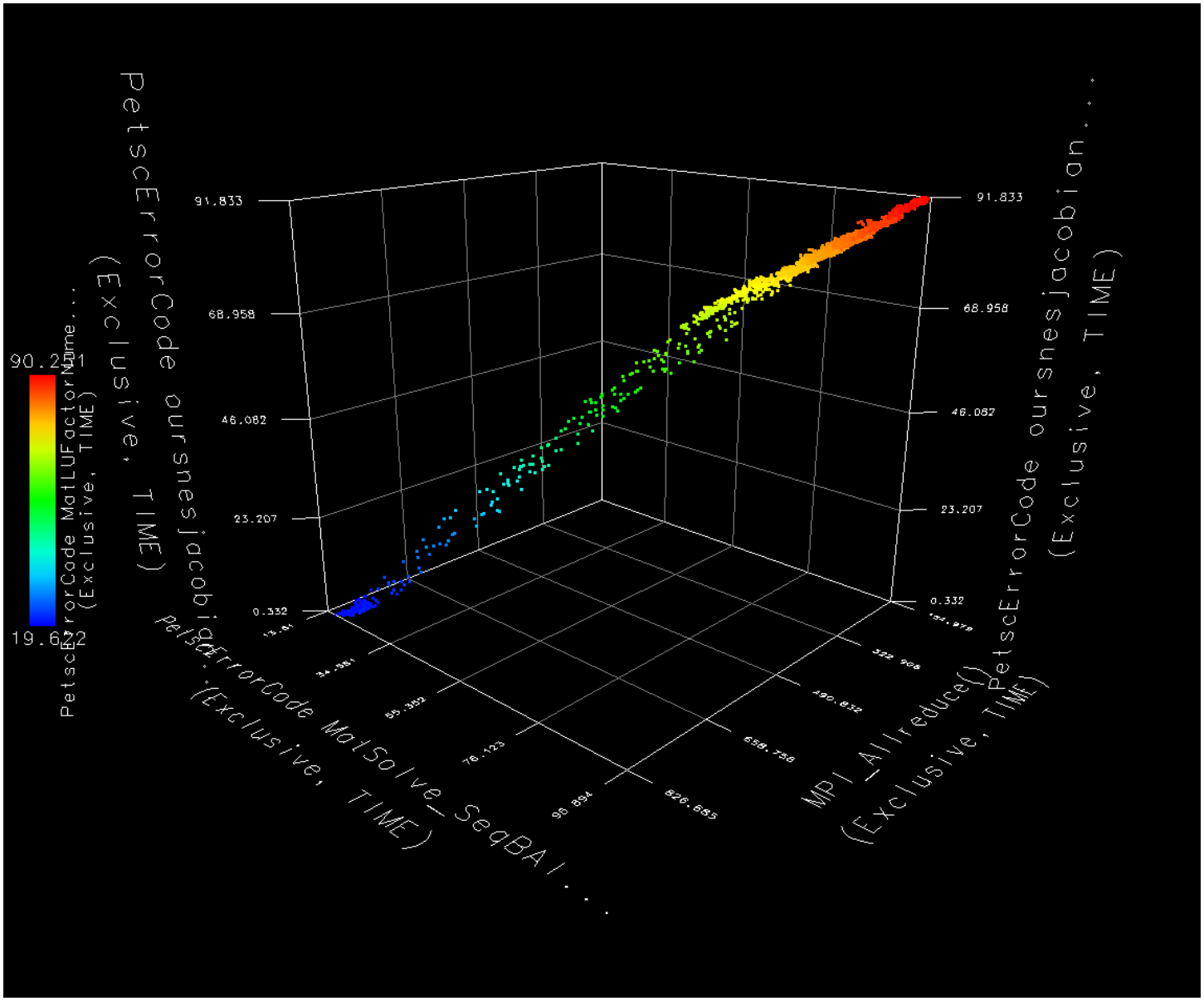
ParaProf call graph



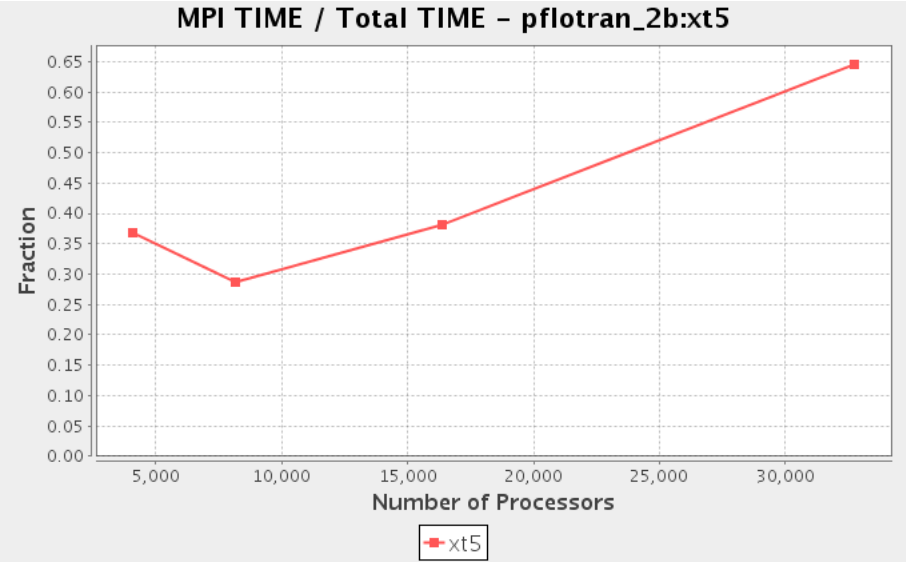
ParaProf mean profile (8184)

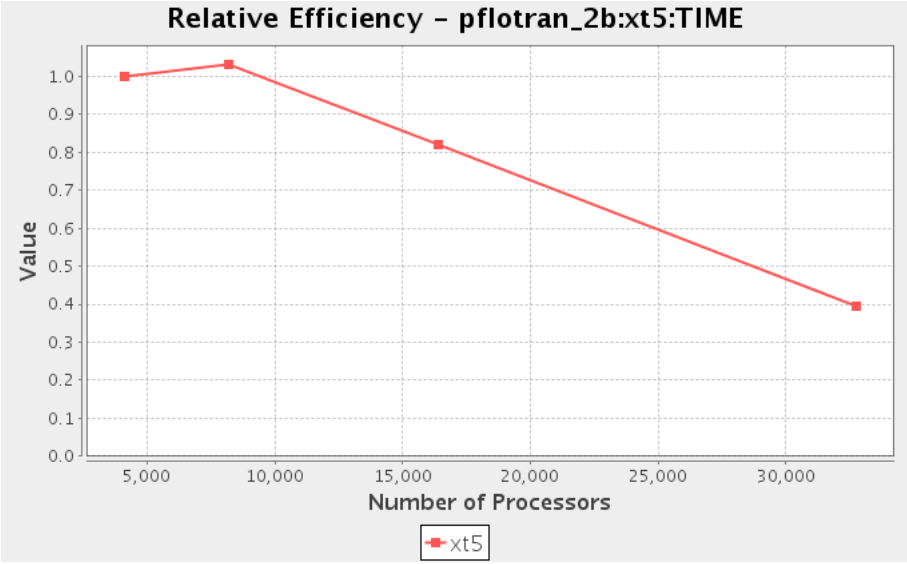


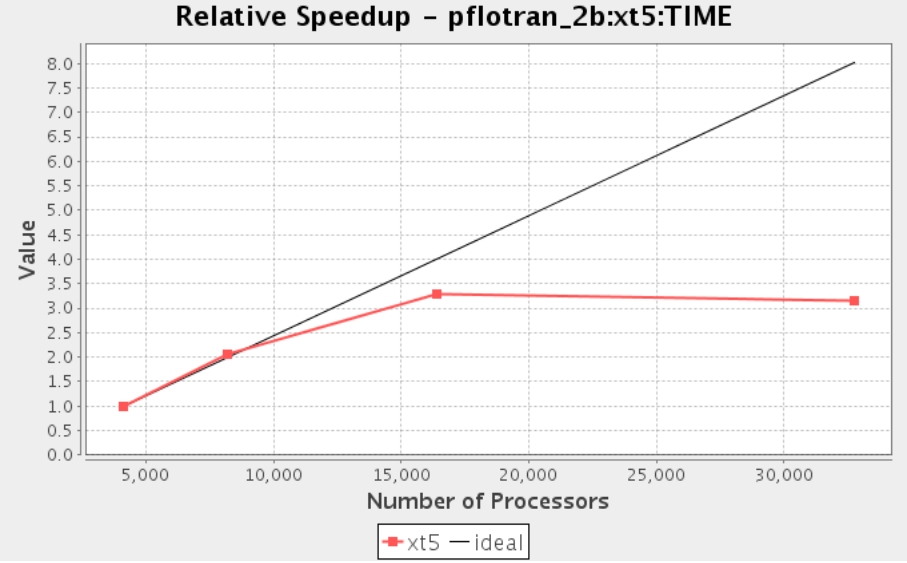
ParaProf 3D correlation cube



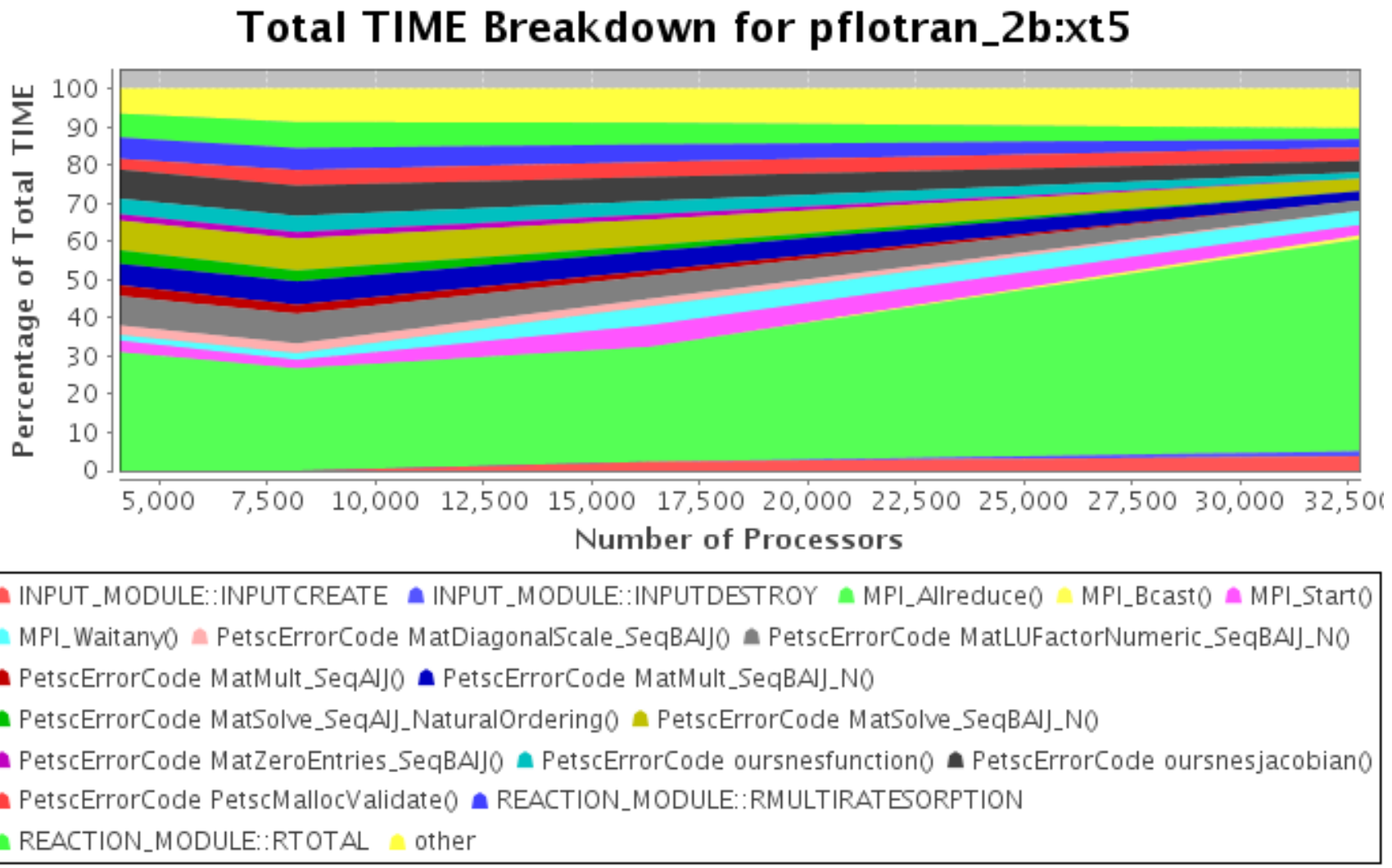
PerfExplorer charts



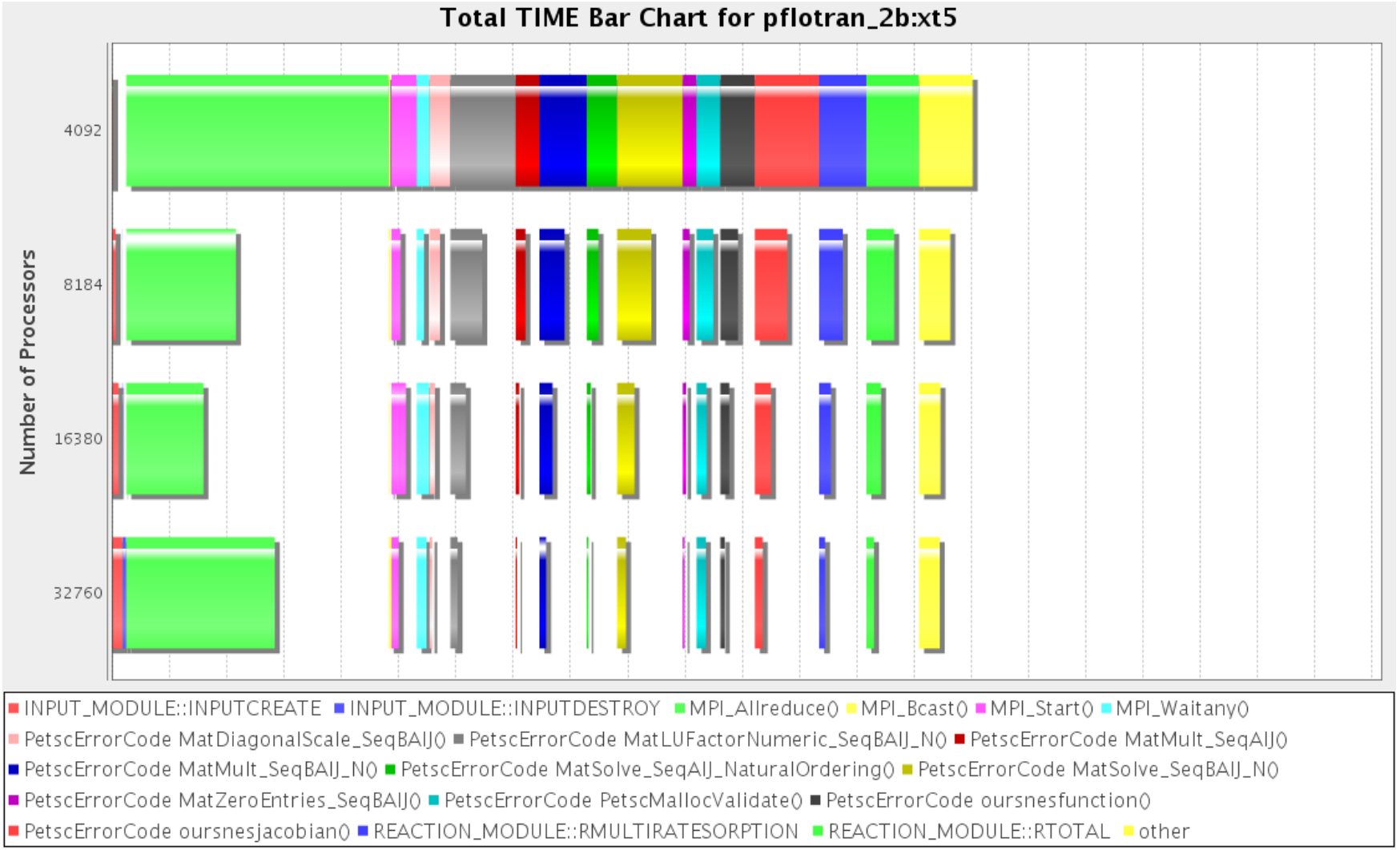




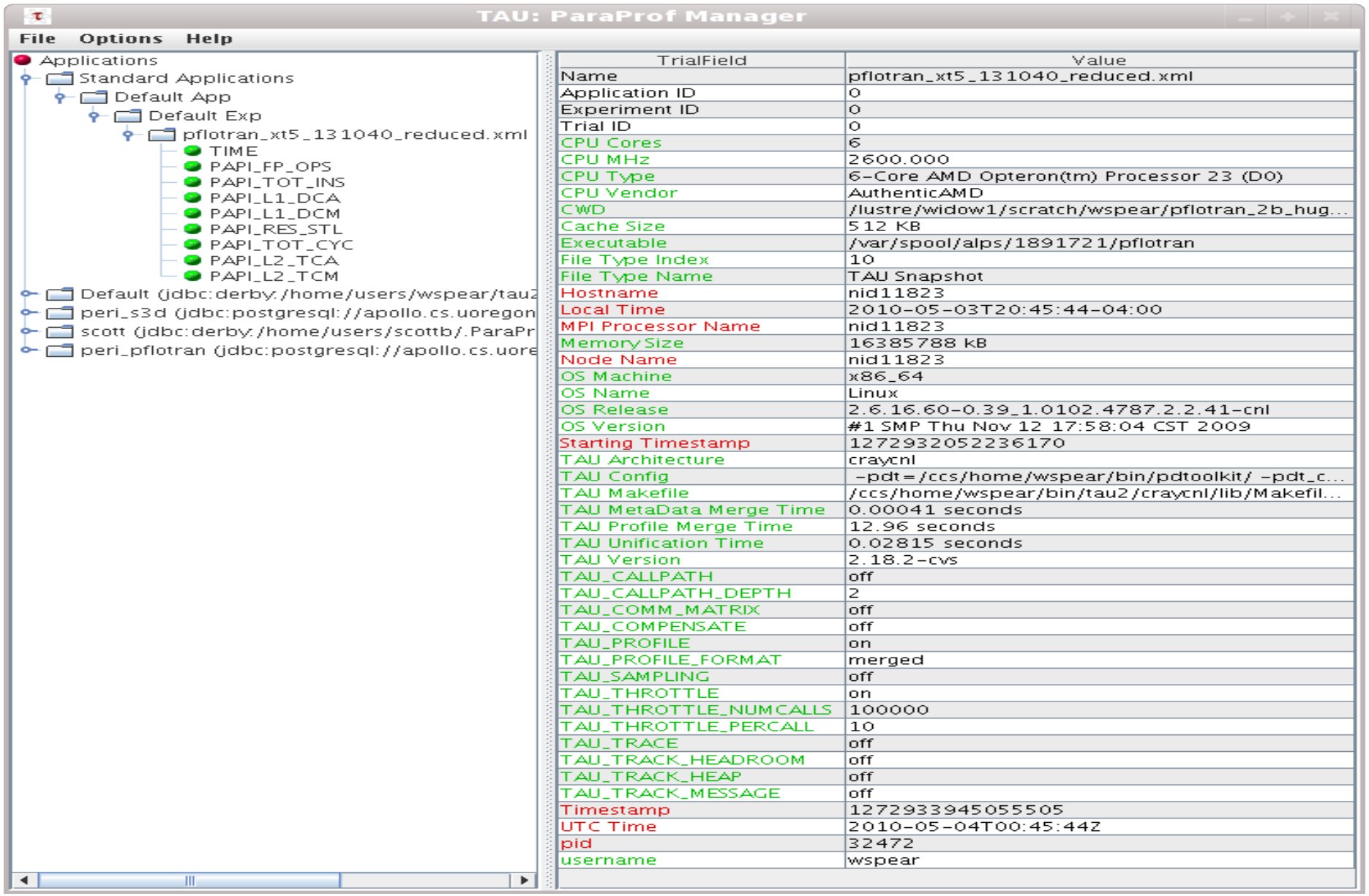
Normalized runtime breakdown



PerfExplorer aligned bar chart

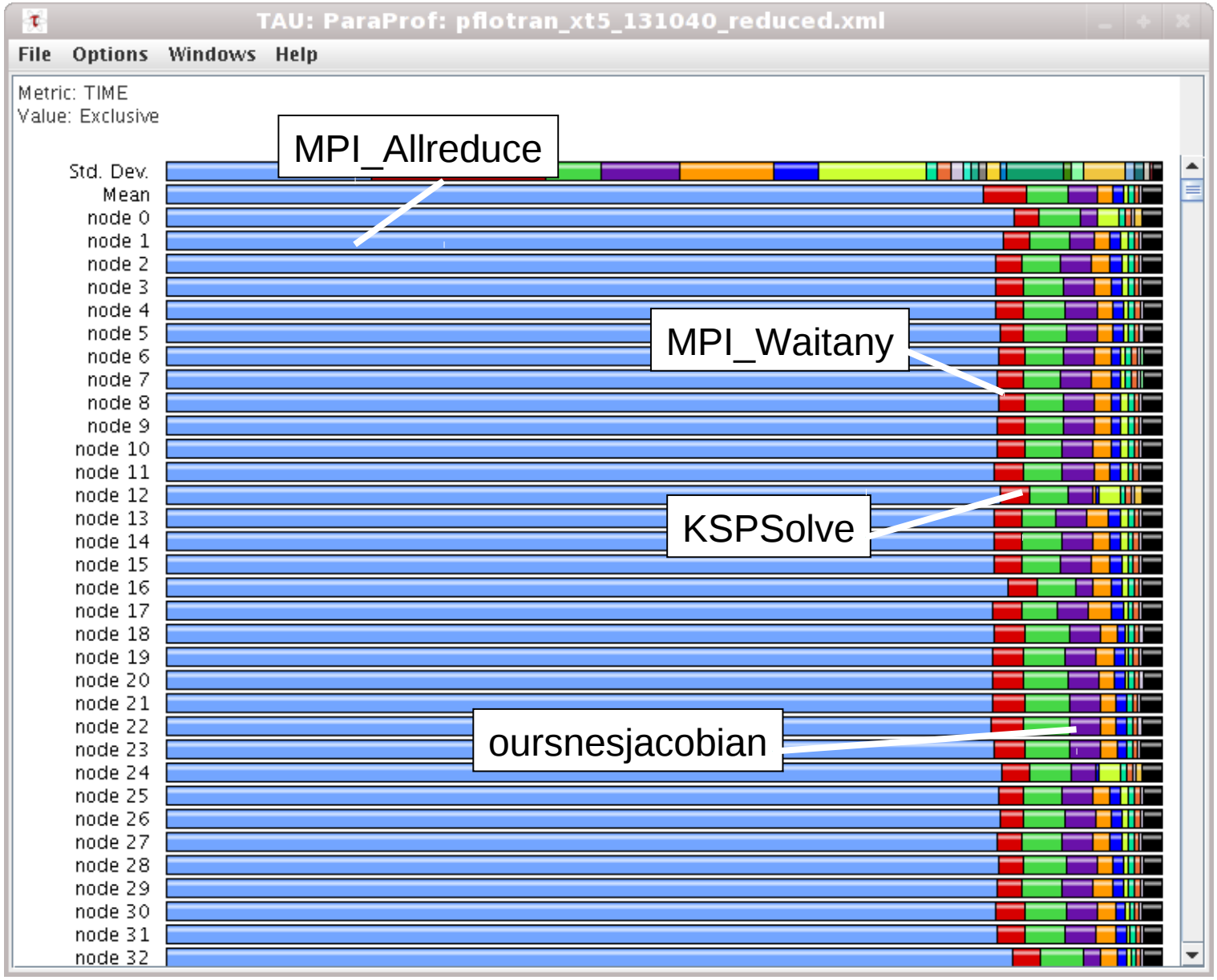


ParaProf Manager (131040)

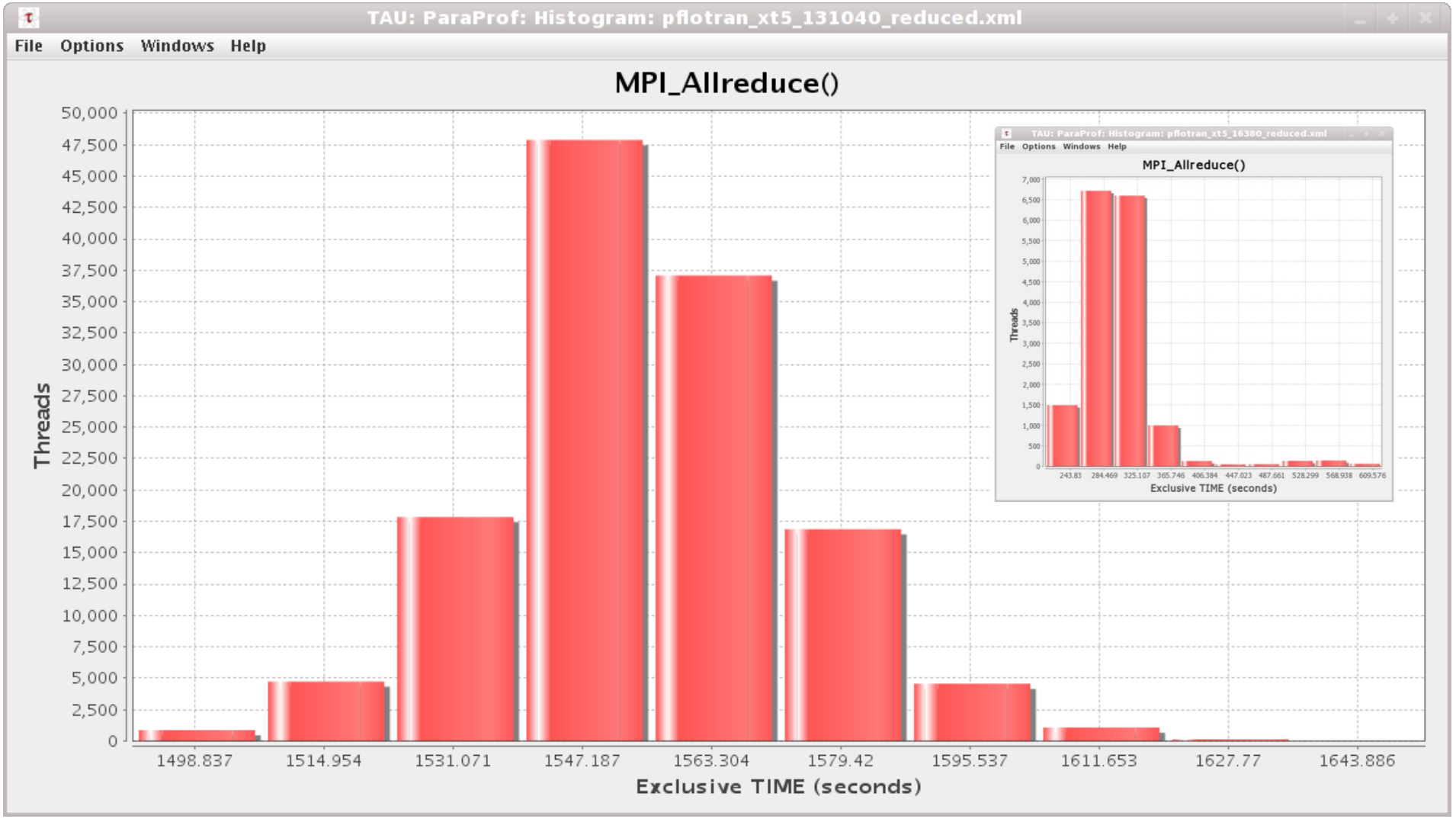


TrialField	Value
Name	pflotran_xt5_131040_reduced.xml
Application ID	0
Experiment ID	0
Trial ID	0
CPU Cores	6
CPU MHz	2600.000
CPU Type	6-Core AMD Opteron(tm) Processor 23 (D0)
CPU Vendor	AuthenticAMD
CWD	/lustre/widow1/scratch/wspear/pflotran_2b_hug...
Cache Size	512 KB
Executable	/var/spool/alps/1891721/pflotran
File Type Index	10
File Type Name	TAU Snapshot
Hostname	nid11823
Local Time	2010-05-03T20:45:44-04:00
MPI Processor Name	nid11823
Memory Size	16385788 kB
Node Name	nid11823
OS Machine	x86_64
OS Name	Linux
OS Release	2.6.16.60-0.39_1.0102.4787.2.2.41-cnl
OS Version	#1 SMP Thu Nov 12 17:58:04 CST 2009
Starting Timestamp	1272932052236170
TAU Architecture	craycnl
TAU Config	-pdt=/ccs/home/wspear/bin/pdtoolkit/ -pdt_c...
TAU Makefile	/ccs/home/wspear/bin/tau2/craycnl/lib/Makefil...
TAU MetaData Merge Time	0.00041 seconds
TAU Profile Merge Time	12.96 seconds
TAU Unification Time	0.02815 seconds
TAU Version	2.18.2-cvs
TAU_CALLPATH	off
TAU_CALLPATH_DEPTH	2
TAU_COMM_MATRIX	off
TAU_COMPENSATE	off
TAU_PROFILE	on
TAU_PROFILE_FORMAT	merged
TAU_SAMPLING	off
TAU_THROTTLE	on
TAU_THROTTLE_NUMCALLS	100000
TAU_THROTTLE_PERCALL	10
TAU_TRACE	off
TAU_TRACK_HEADROOM	off
TAU_TRACK_HEAP	off
TAU_TRACK_MESSAGE	off
Timestamp	1272933945055505
UTC Time	2010-05-04T00:45:44Z
pid	32472
username	wspear

ParaProf full profile (131040)



ParaProf histogram (131040)



Review of PFLOTRAN analyses

- Duplication of MPI Communicators by HDF5 dominates initialization at scale (particularly on XT5)
- MPI_Allreduce collective communication remains a severe bottleneck in the timestep loop
  - for both FLOW & TRAN phases
  - particularly MatZeroRowsLocal/PetscMaxSum
- Computational imbalance from inactive grid cells evident for processes at one end of 3D process grid
  - widespread in the computational routines
  - since only a minority of processes, little to gain

Caveat emptor

- Performance analysis of complex applications at large-scale requires care
  - full automatic instrumentation is convenient, but may produce more detail than desirable
  - selective measurement and/or instrumentation may be used to reduce overhead and size of event traces
  - even basic reduced execution profiles for many thousands of processes rapidly become awkwardly large
  - powerful analyses and interactive customizable visualizations required for an effective initial overview leading to in-depth refinement of performance issues
- VI-HPS experts are available to assist

Summary

- Applications can be prepared for measurement using manual, selective and automatic instrumentation and PAPI counters
- Scalasca runtime summary & automatic event trace analyses quantify and isolate locations of MPI communication & synchronization overheads
- Vampir visual trace analysis supports interactive exploration of detailed process interaction and clustering of similar traces
- TAU experiment management, graphical profile displays and extensive instrumentation options facilitate customization of measurements and analysis presentations
- VI-HPS tools can be used independently, however, maximum benefit offered by exploiting their complementary integrated capabilities

Acknowledgements

- PFLOTRAN developers for providing an interesting large-scale test case and assistance building/running their code
- Dagstuhl seminar organizers for setting the tools challenge and hosting the associated discussions
- NIC/JSC & NCCS/ORNL for use of compute-time allocations on their large-scale facilities
- Support and development teams for the various tools for their prompt support