



Performance analysis: Paraver hands-on using the HARMONIE model

Xavier Yepes-Arbós
Mario C. Acosta

27/8/2020

Introduction

- In this session we will show the basics of Paraver to analyze a trace taking into account different aspects:
 - Load balance
 - Communication
 - Computation
- Using Paraver we will have both a global qualitative perception of the application and a detailed quantitative analysis of it.
- The trace was obtained from a small execution of the HARMONIE weather forecast model.
- HARMONIE was ran on the ECMWF's Cray XC40 system based on Intel chips.
- It used 285 MPI processes and without OpenMP.

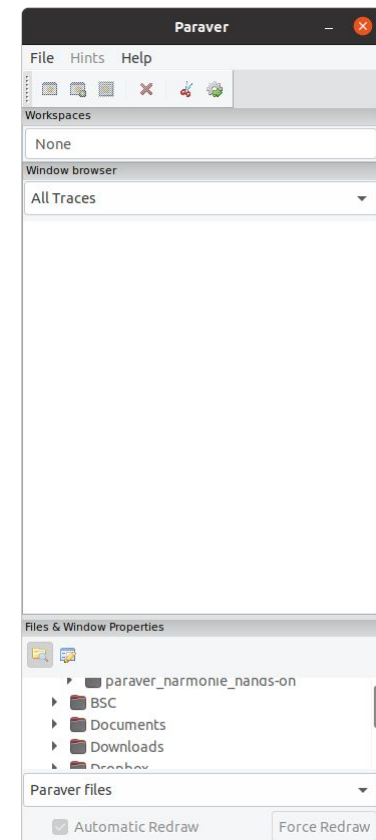
What is a trace made of?

- A trace is made of three files: *.prv, *.pcf and *.row
- For this hands-on we have:
 - harmonie_285MPI_noopenmp.pcf: The trace file itself. It contains the performance events.
 - harmonie_285MPI_noopenmp.pcf: It describes the type of performance events contained in the .prv file.
 - harmonie_285MPI_noopenmp.row: It contains the ids of the MPI processes and the OpenMP threads.
- In particular, if you search for EVENT_TYPE in the pcf file, you will see what the trace contains exactly. For example:

```
EVENT_TYPE
9  50000001  MPI Point-to-point
VALUES
33  MPI_Bsend
4   MPI_Irecv
3   MPI_Isend
2   MPI_Recv
1   MPI_Send
5   MPI_Wait
59  MPI_Waitany
0   Outside MPI
```

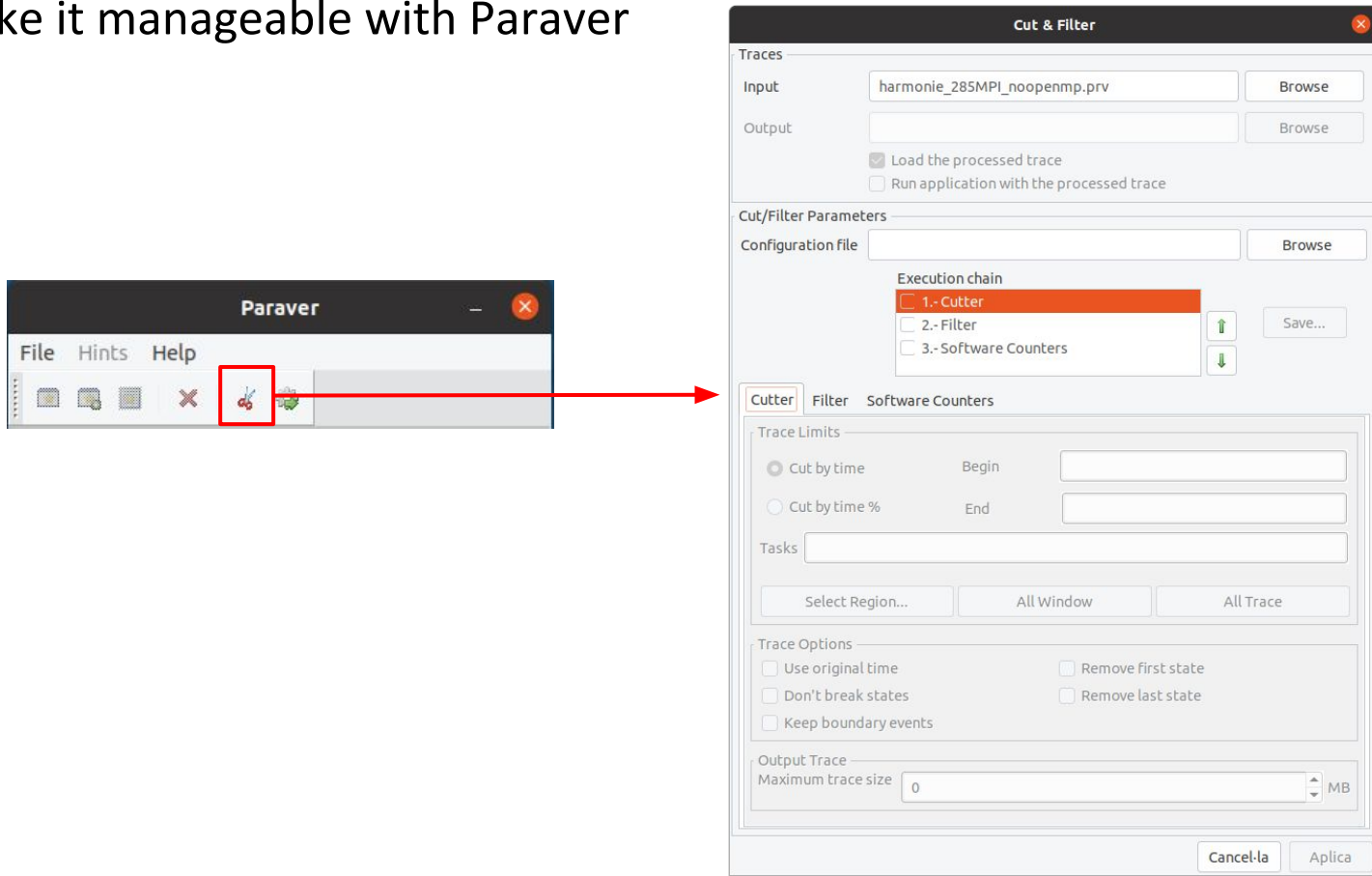
First steps

- Uncompress the tar file using this command:
`tar zxfv supporting_tools-performance_analysis.tar.gz`
- Open Paraver by executing:
`wxparaver&`
- The Paraver window should pop up such as this one:



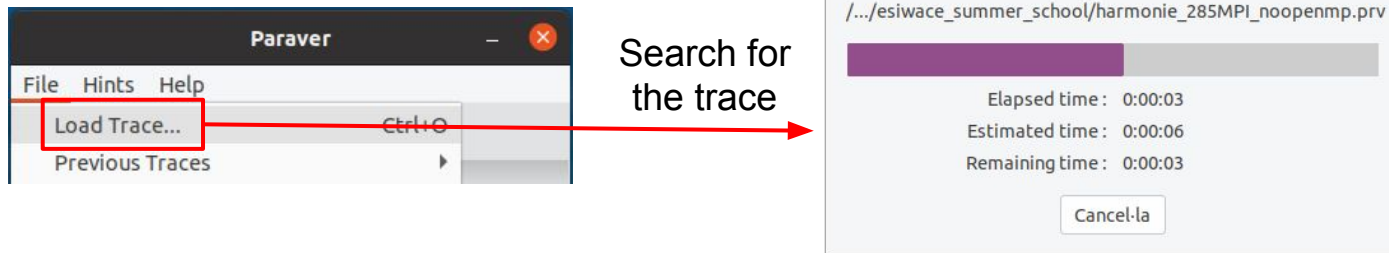
Cut & filter

In case of a trace is very big (for instance > 1 GB), it might be cut and filtered to make it manageable with Paraver

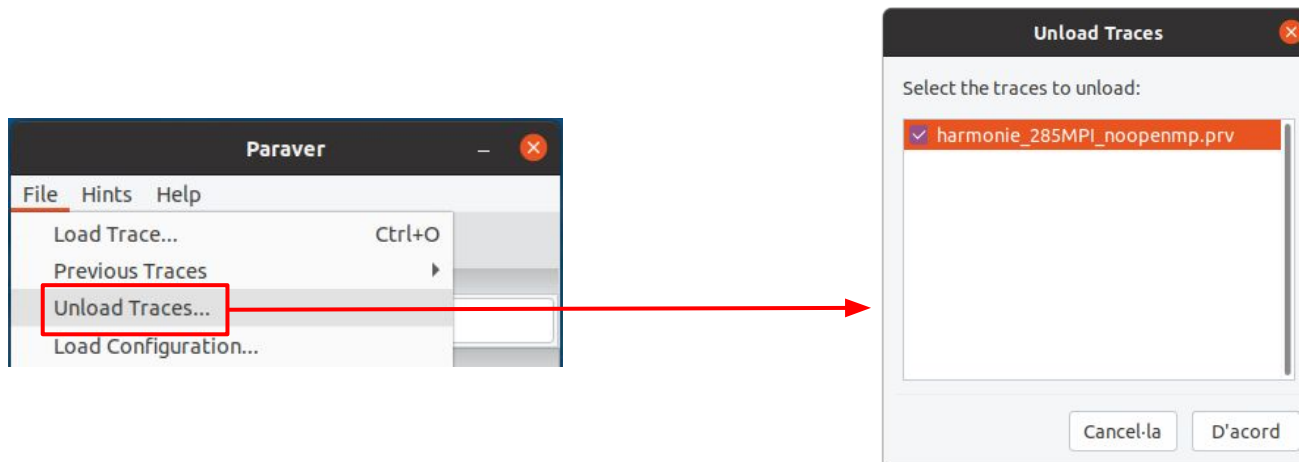


(Un)load a trace

- “File > Load Trace...” and search for the `harmonie_285MPI_noopenmp.prv` file.

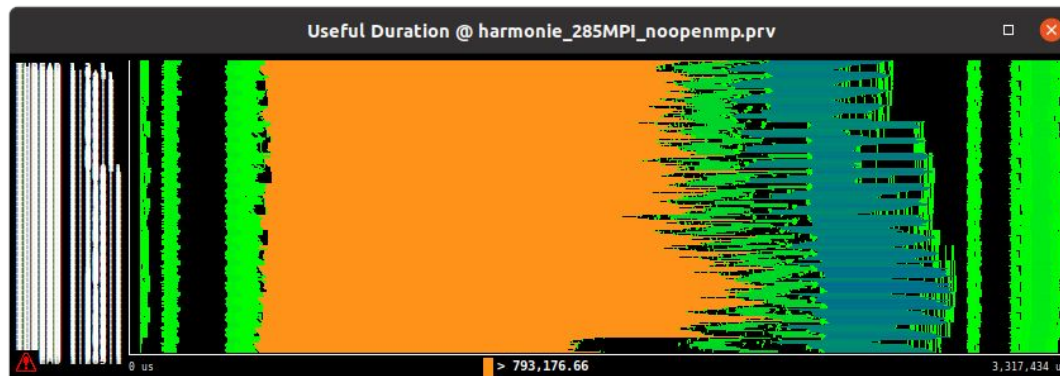
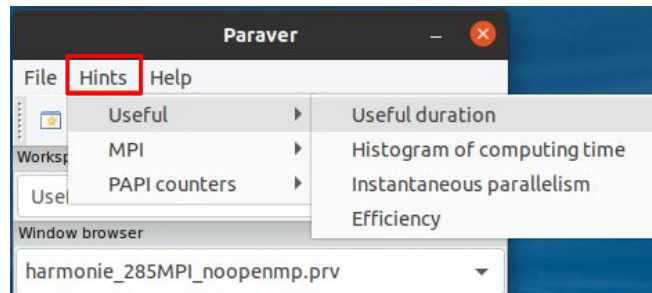


- “File > Unload Traces...” and select the trace that you want to unload.



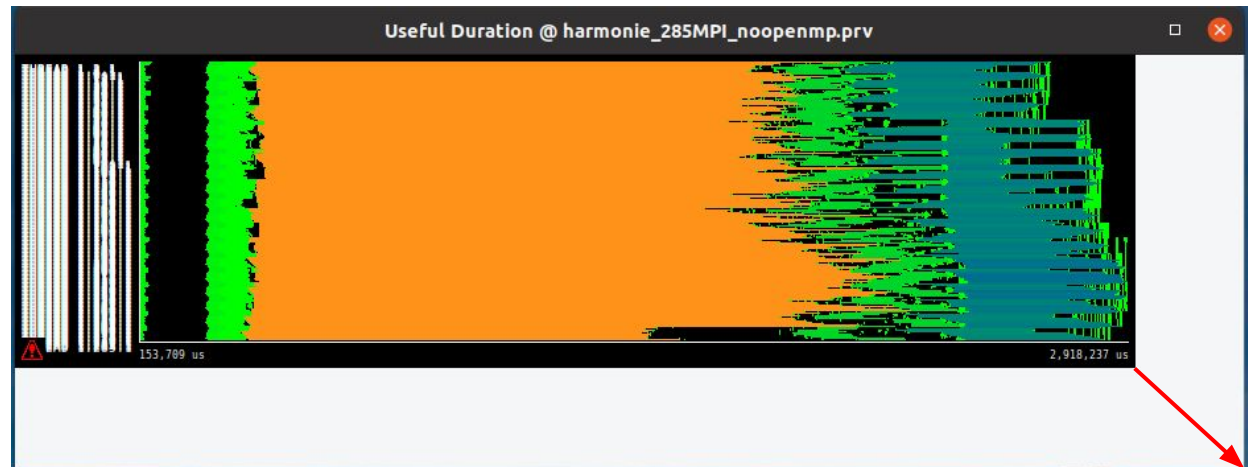
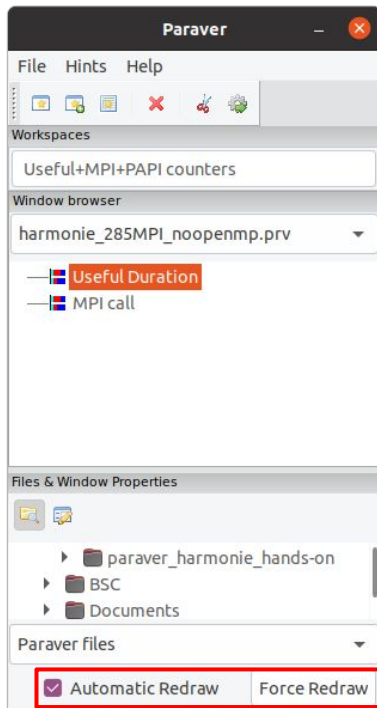
Use of Paraver hints

- Paraver provide some hints which are views and histograms of basic performance metrics included in the trace.
- In “Hints”, feel free to load whatever option you want.



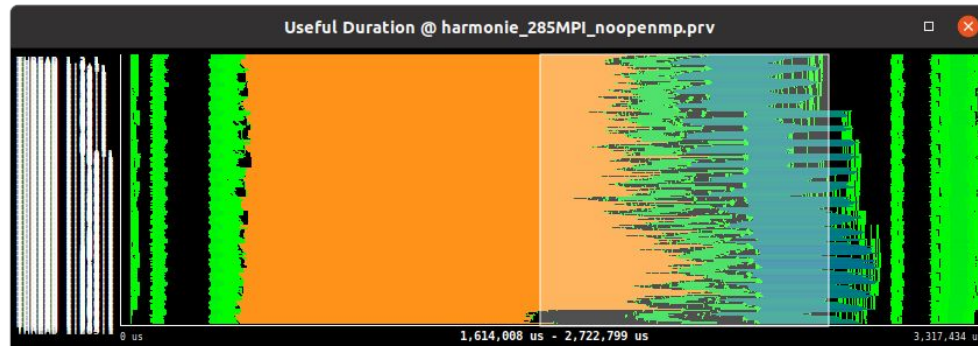
Basics of Paraver: resize

When you resize the window of a view, Paraver automatically redraws it. Depending on the situation, this might be useful (such in our case), or it might be a problem due to the time of rendering the view again.

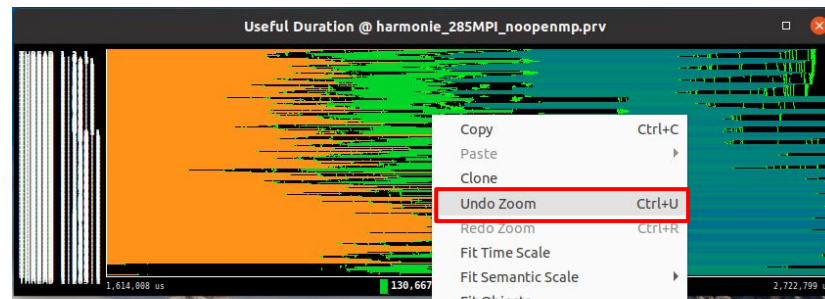


Basics of Paraver: (undo) zoom

- To make zoom into a period of time, just click on the desired area. If you also interested on zooming in MPI processes, press Ctrl while clicking the desired area.

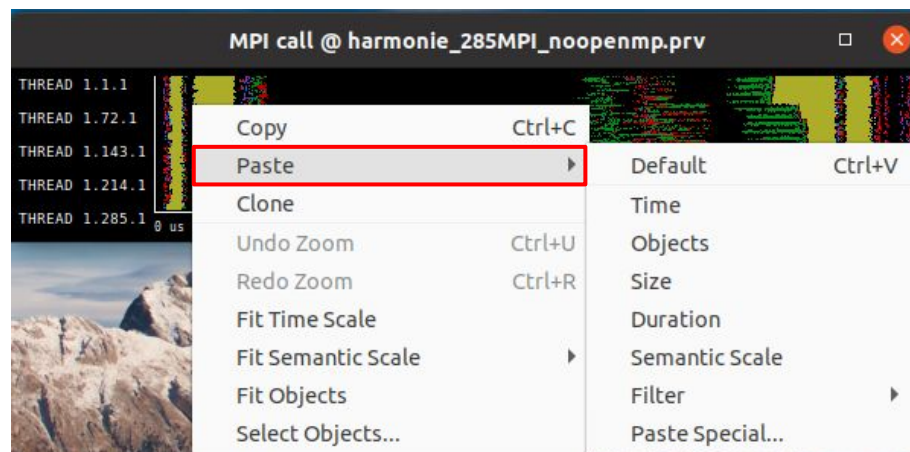
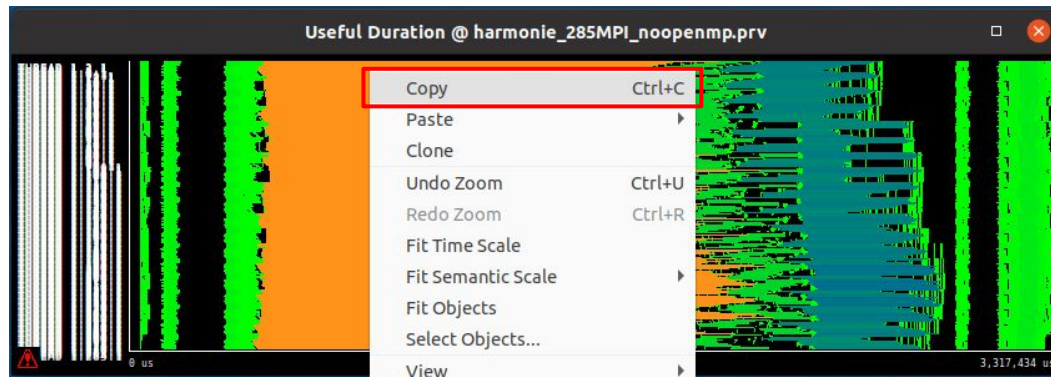


- To undo the zoom, right click on the window and “Undo Zoom”



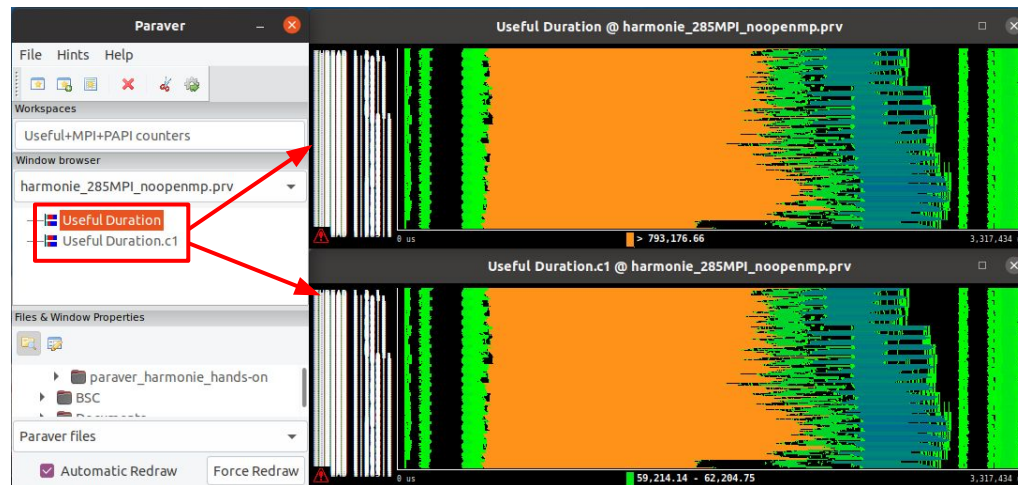
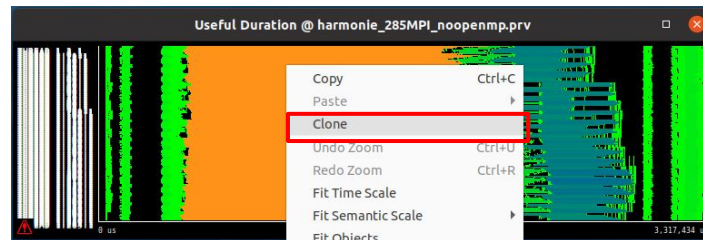
Basics of Paraver: copy & paste

- It is possible to copy and paste duration, time, size, etc between windows.
- To do so, right click on the window and “Copy” or “Paste > X”.



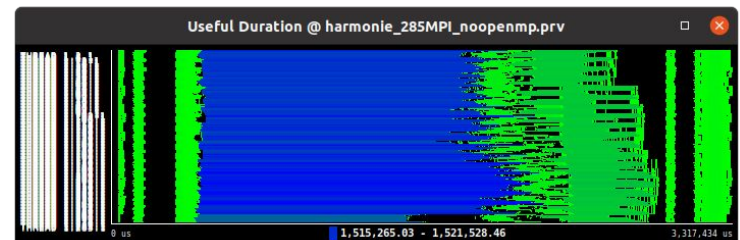
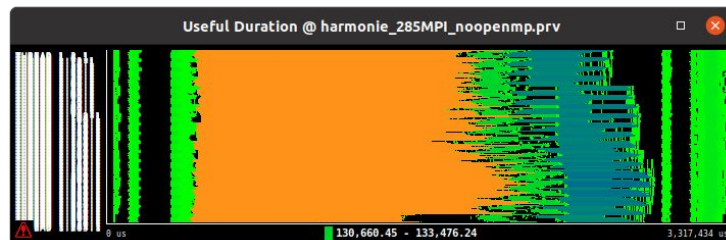
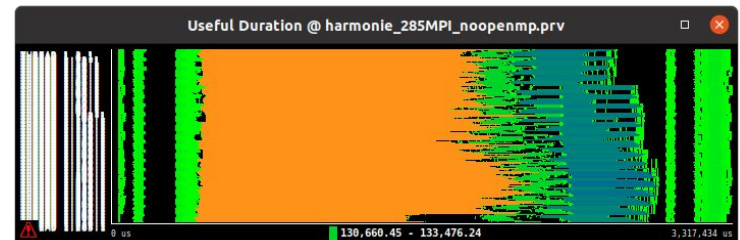
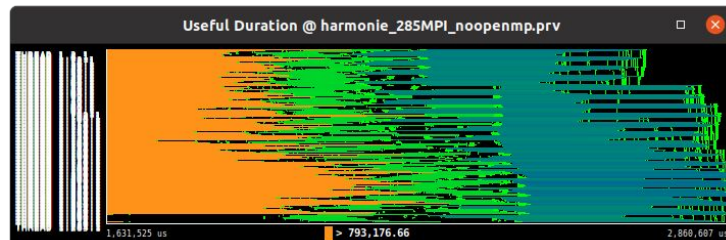
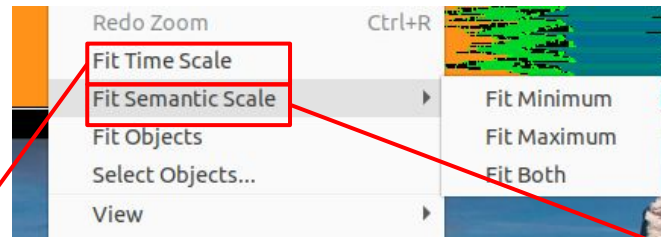
Basics of Paraver: clone

- In some situations, it might be useful to clone the same window to try different things.
- To do so, right click on the window and “Clone”



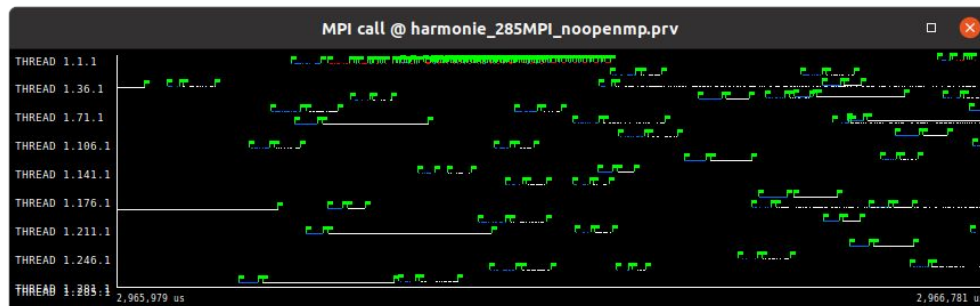
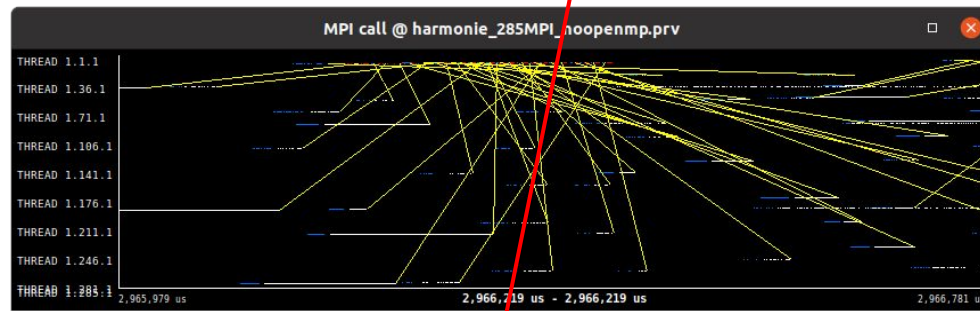
Basics of Paraver: time and semantic scales

It is useful to adjust both the time scale (x axis) and the semantic scale (colors of the window). The latter is important to paint events taking into account the minimum and maximum values.



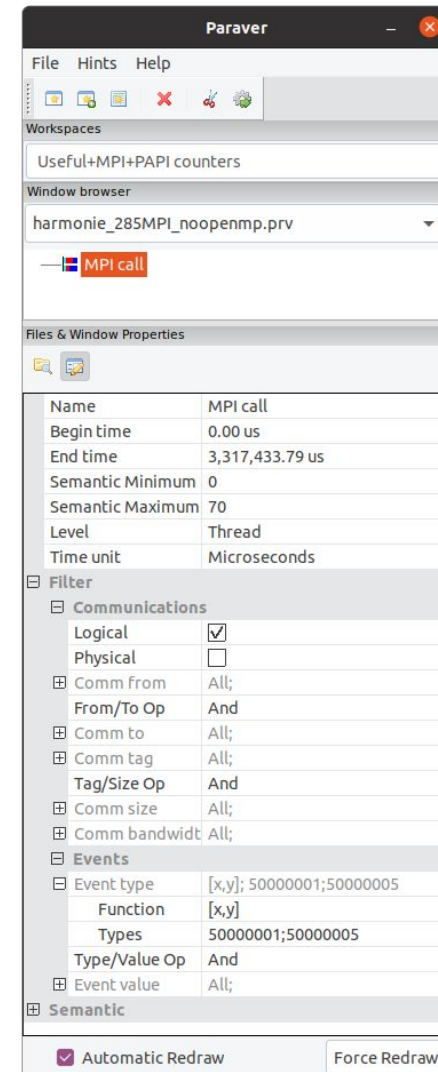
View (communication lines and event flags)

There are two interesting options to draw the communications lines and/or the event flags.



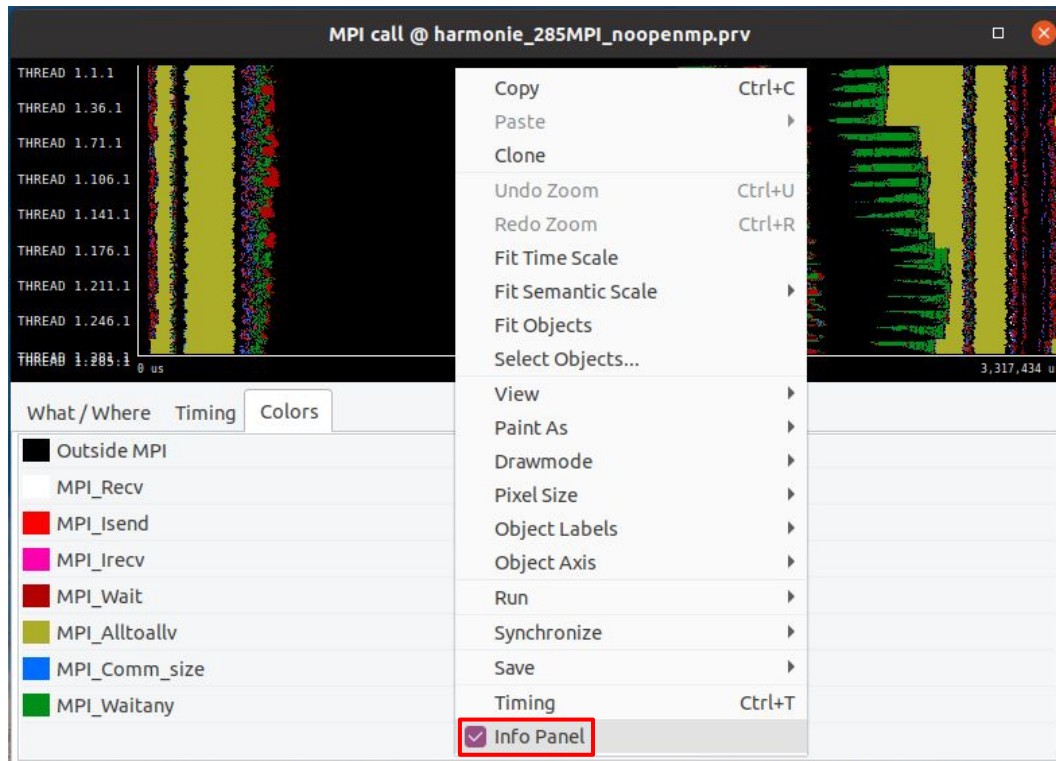
Window properties section

- In the Paraver window, the Window Properties section gives detailed information about what it is represented in the current view and how.
- It allows to change different values, such as the time scale, the semantic scale, which communications to draw, what types of events, etc.



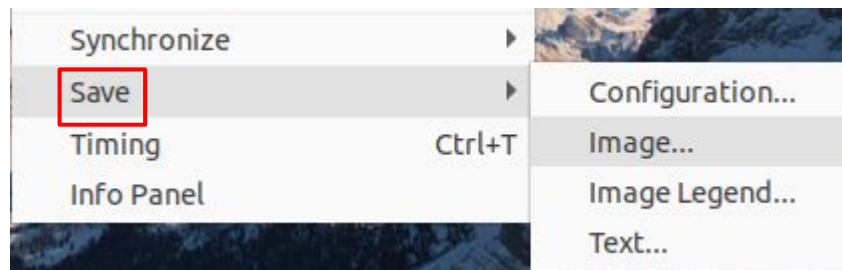
Info panel

- An Info Panel might be enabled by right-clicking on the window and choosing “Info Panel”.
- It contains the legend of the window, timing, duration, etc.



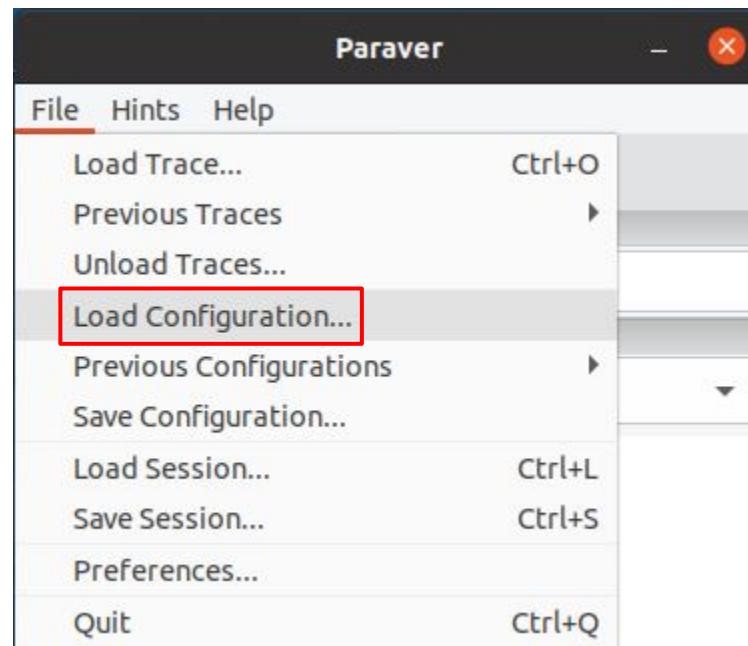
Save a snapshot

- In case you are interested in saving the view of a window, it is possible to save it.
- To do so, right-click on the window and “Save > Image...”.



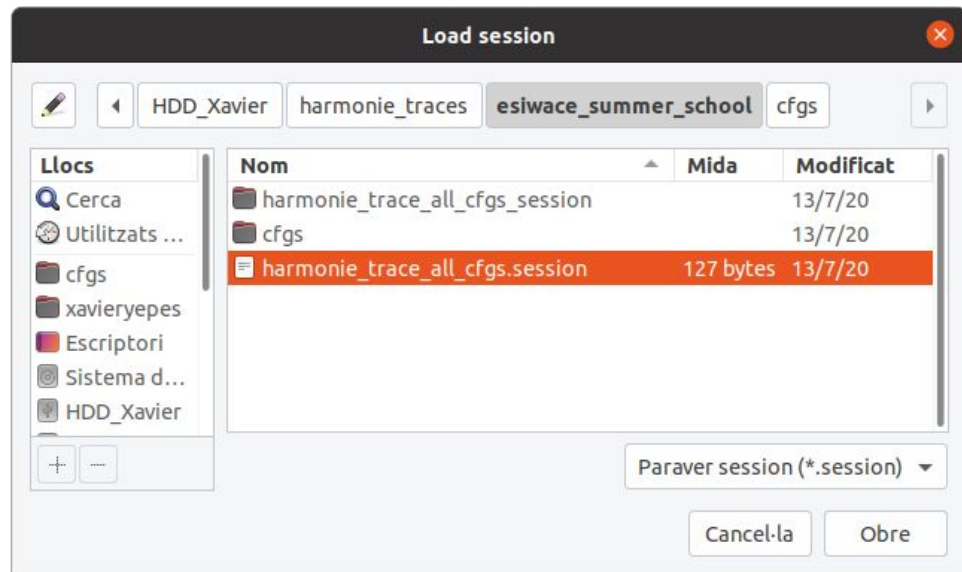
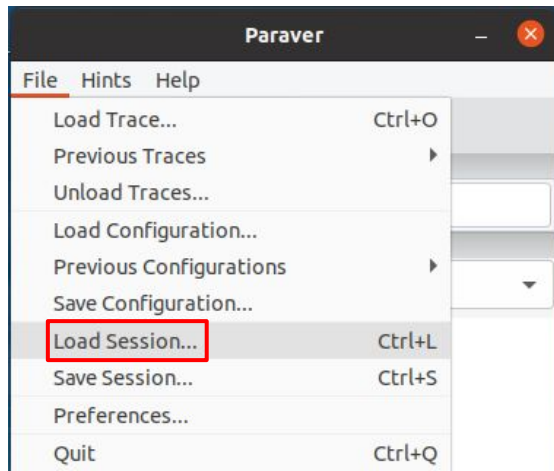
Load a configuration

- Paraver allows to study a lot of different performance metrics by loading configuration files (.cfg) that describe how to do it.
- To load a specific configuration file, go to “File > Load Configuration...” and search for the file that you want to load.



Load/save a session

- Sometimes you might have many different views open, so you would like to save this “environment” to continue working on it in another moment.
- It is possible to do it by using sessions.
- To load a session, go to “File > Load Session...” and search for the session.
- To save an ongoing session, go to “File > Save Session...” and save it.

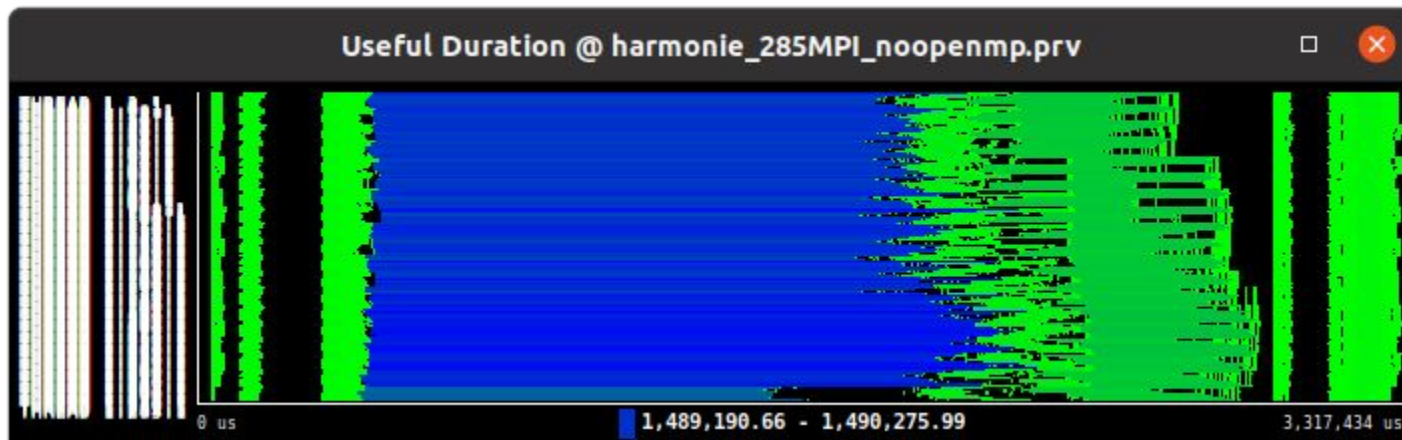


Notes

- The following slides will show views of the different configuration files used for the hands-on.
- You can load the configuration files one by one, or you can just load the session of the hands-on which will load all the configuration files automatically.

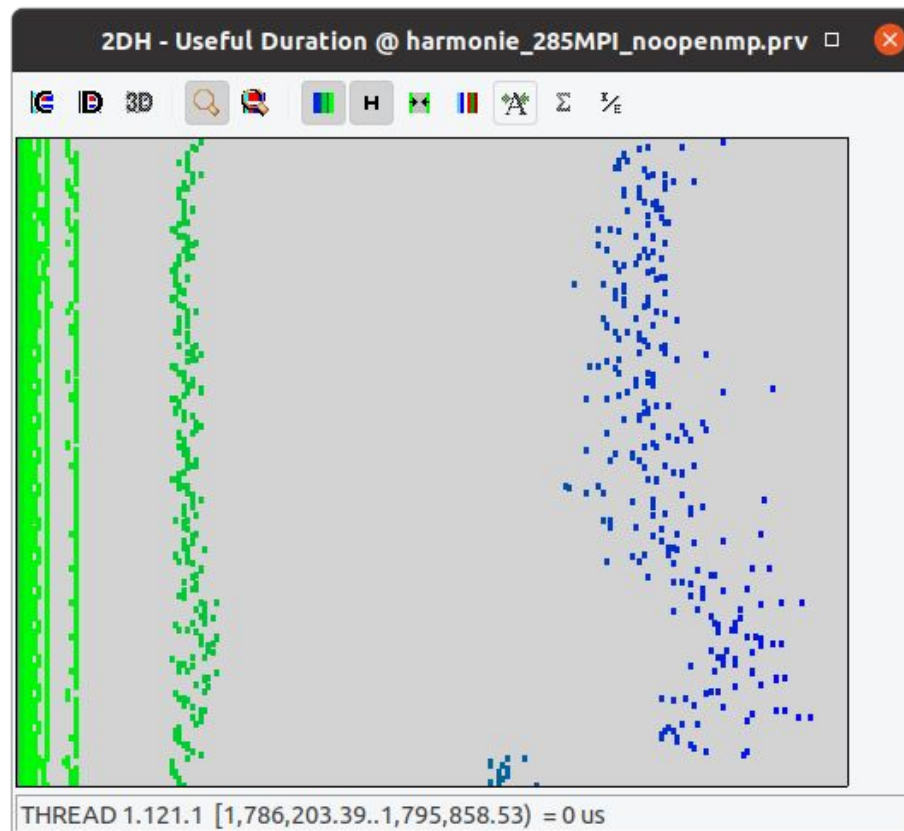
Useful duration

useful_duration.cfg: It shows the duration of the computational chunks.



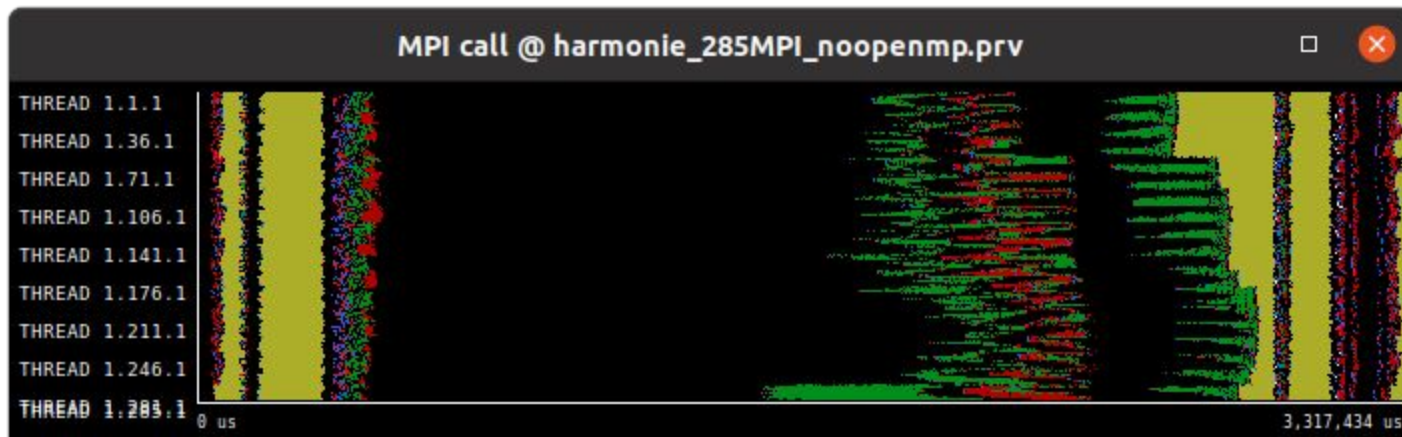
Histogram of useful duration

useful_duration_histogram.cfg: It shows a histogram of the duration of the computational chunks to study the load balance.



MPI calls

mpi_call.cfg: It shows the different calls to MPI functions along the execution.



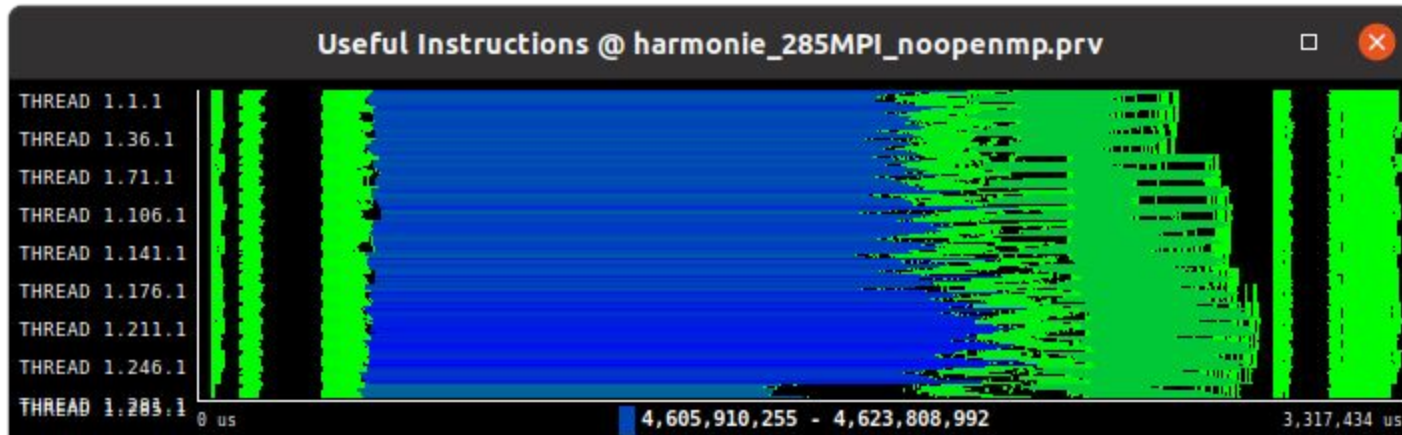
MPI profile

mpi_stats.cfg: It shows the MPI profile to know the load balance, parallel efficiency, etc.

MPI call profile @ harmonie_285MPI_noopenmp.prv								
	Outside MPI	MPI_Recv	MPI_Isend	MPI_Irecv	MPI_Wait	MPI_Alltoallv	MPI_Comm_size	MPI_Waitany
Total	20,751.31 %	2.16 %	44.77 %	13.23 %	1,058.01 %	4,197.54 %	11.15 %	2,421.84 %
Average	72.81 %	0.01 %	0.16 %	0.05 %	3.71 %	14.73 %	0.04 %	8.50 %
Maximum	83.79 %	0.08 %	0.91 %	0.18 %	11.58 %	19.93 %	0.05 %	23.27 %
Minimum	57.97 %	0.00 %	0.02 %	0.01 %	0.65 %	11.23 %	0.03 %	1.03 %
StDev	5.00 %	0.01 %	0.13 %	0.03 %	2.04 %	2.48 %	0.00 %	3.86 %
Avg/Max	0.87	0.11	0.17	0.26	0.32	0.74	0.82	0.37

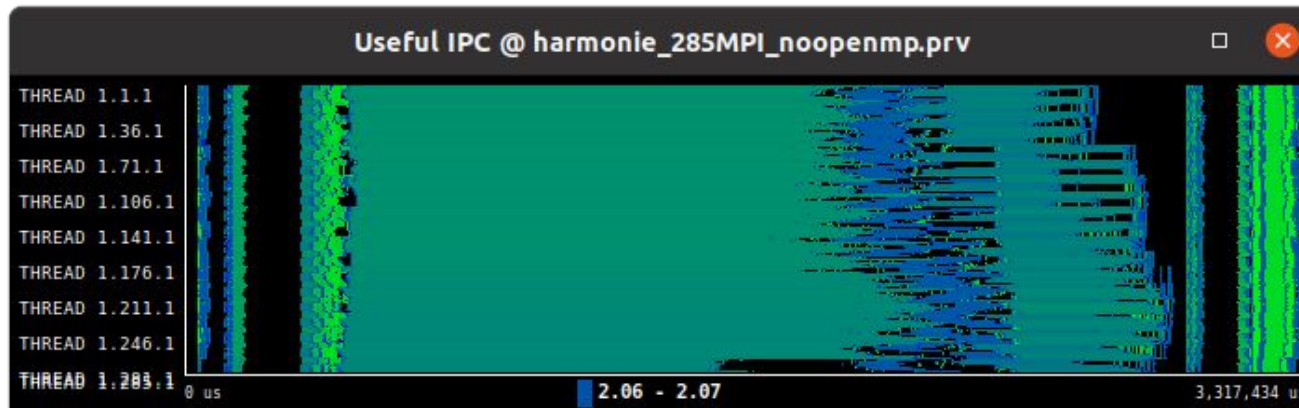
Useful instructions

useful_instructions.cfg: It shows the executed number of useful instructions.

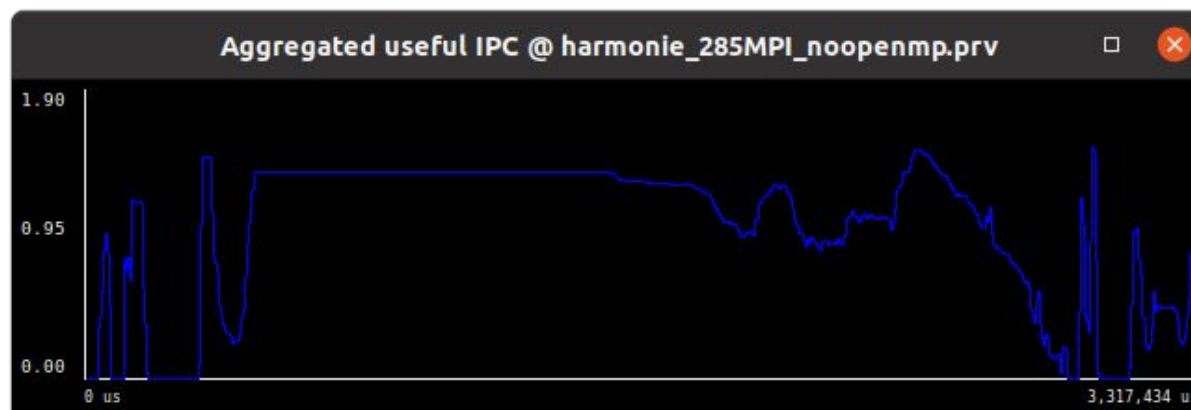


IPC

- `useful_ipc.cfg`: It shows the useful Instructions per Cycle (IPC).

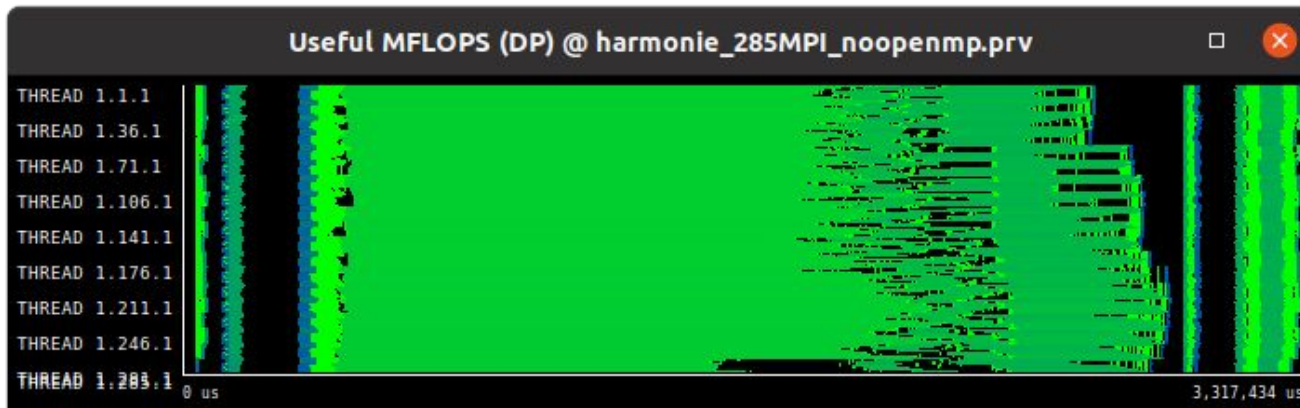


- `aggregated_useful_ipc.cfg`: It shows the aggregated useful IPC.

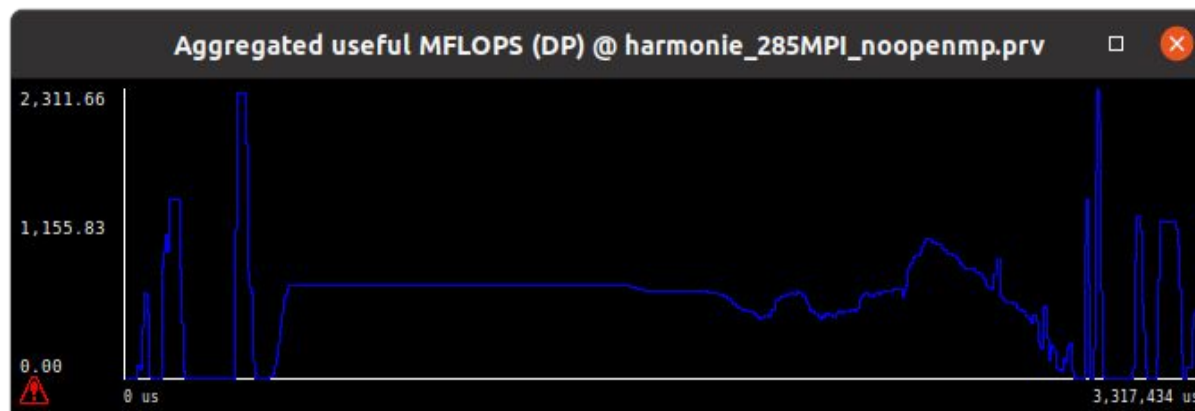


MFLOPS

- useful_mflops.cfg: It shows the useful MFLOPS

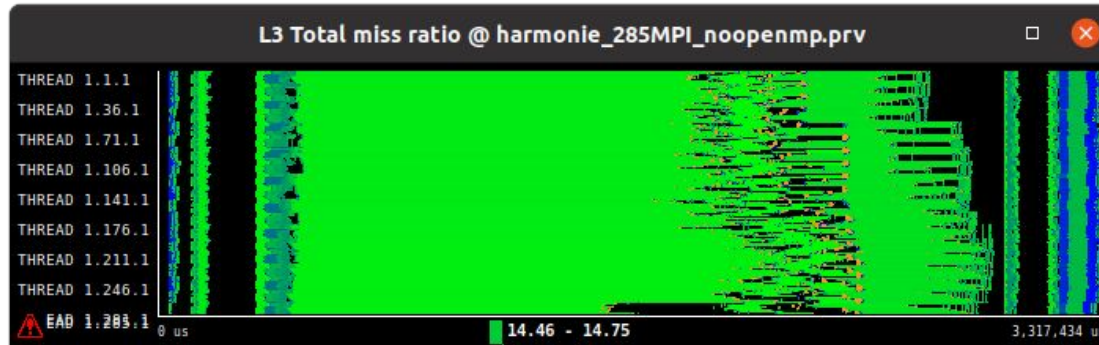


- aggregated_useful_mflops.cfg: It shows the aggregated useful MFLOPS.

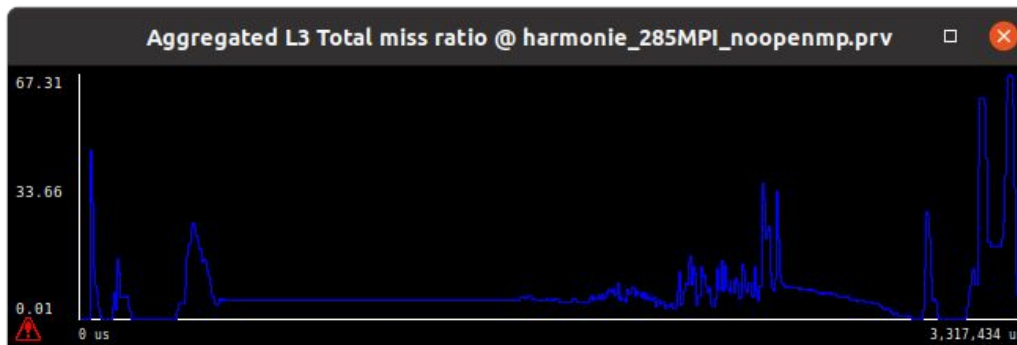


L3 cache misses

- `useful_L3_misses_per_1000_instructions.cfg`: It shows the ratio of L3 cache misses per 1000 useful instructions.



- `aggregated_useful_L3_misses_per_1000_instructions.cfg`: It shows the aggregated ratio of L3 cache misses per 1000 useful instructions.

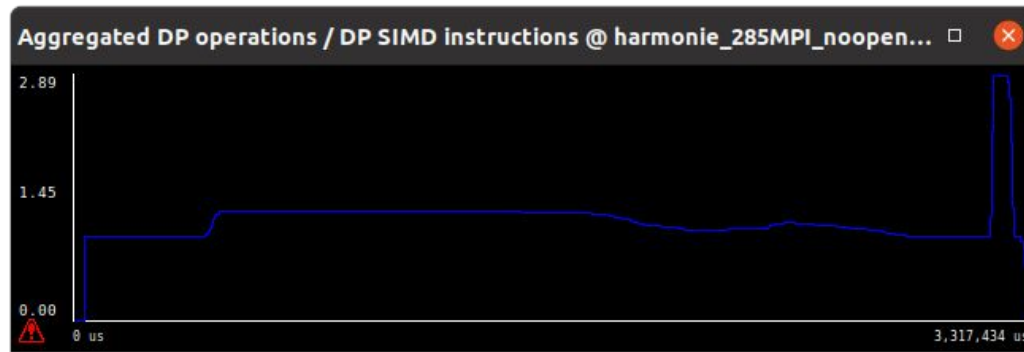


Vectorization efficiency

- `useful_double-precision_vectorization_ratio.cfg`: It shows the percentage of double precision floating-point operations per SIMD instruction.



- `aggregated_useful_double-precision_vectorization_ratio.cfg`: It shows the aggregated percentage of double precision floating-point operations per SIMD instruction.



Bonus

- If you feel encouraged to explore more options of Paraver, feel free to “play” with any functionality.
- We suggest you to explore different configuration files included in the Paraver folder: wxparaver-4.8.2-Linux_x86_64/cfgs
- Note that not all configuration files might be useful for the trace of this hands-on since it depends on the available events in the trace.



**Barcelona
Supercomputing
Center**
Centro Nacional de Supercomputación



**EXCELENCIA
SEVERO
OCHOA**

Thank you



esiwace
CENTRE OF EXCELLENCE IN SIMULATION OF WEATHER
AND CLIMATE IN EUROPE



xavier.yepes@bsc.es