

Hotspots Analysis Intel® VTune™ Profiler

Heinrich Bockhorst

Durham, April 11th 2024



Agenda VTune Hotspots

- Overview on this analysis type: views, filters, grouping
- How to run an analysis GUI vs. Command Line
- How to handle MPI programs
- Demo showing typical steps

Optimize Performance

Intel® VTune™ Profiler

Get the Right Data to Find Bottlenecks

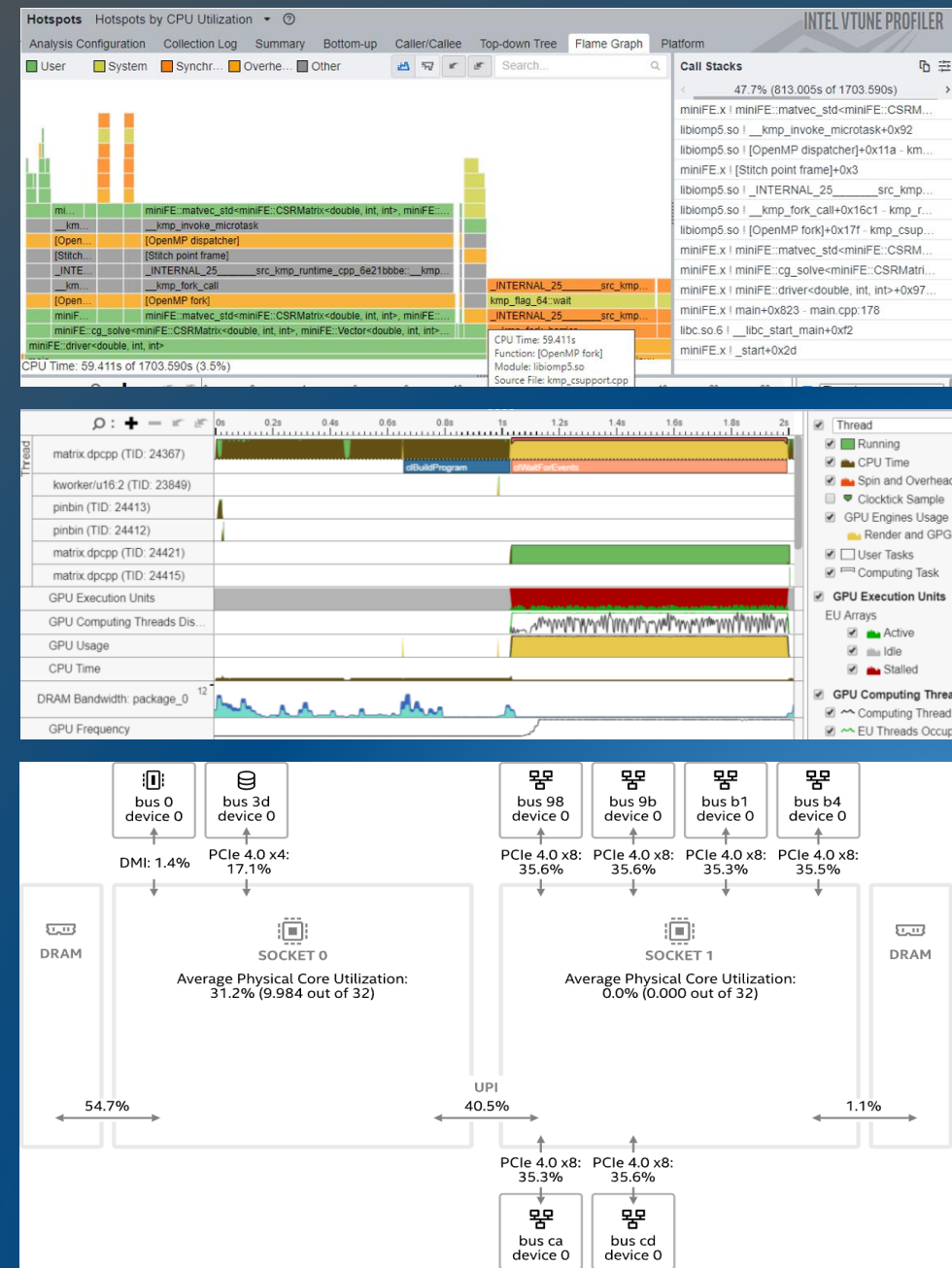
- A suite of profiling for CPU, GPU, FPGA, threading, memory, cache, storage, offload, power...
- Application or system-wide analysis
- DPC++, C, C++, Fortran, Python*, Go*, Java*, or a mix
- Linux, Windows, FreeBSD, Android, Yocto and more
- Containers and VMs

Analyze Data Faster

- Collect data HW/SW sampling and tracing w/o re-compilation
- See results on your source, in architecture diagrams, as a histogram, on a timeline...
- Filter and organize data to find answers

Work Your Way

- User interface or command line
- Profile locally and remotely
- GUI (desktop or web) or command line



Find Answers Fast

Intel® VTune™ Profiler

Adjust Data Grouping

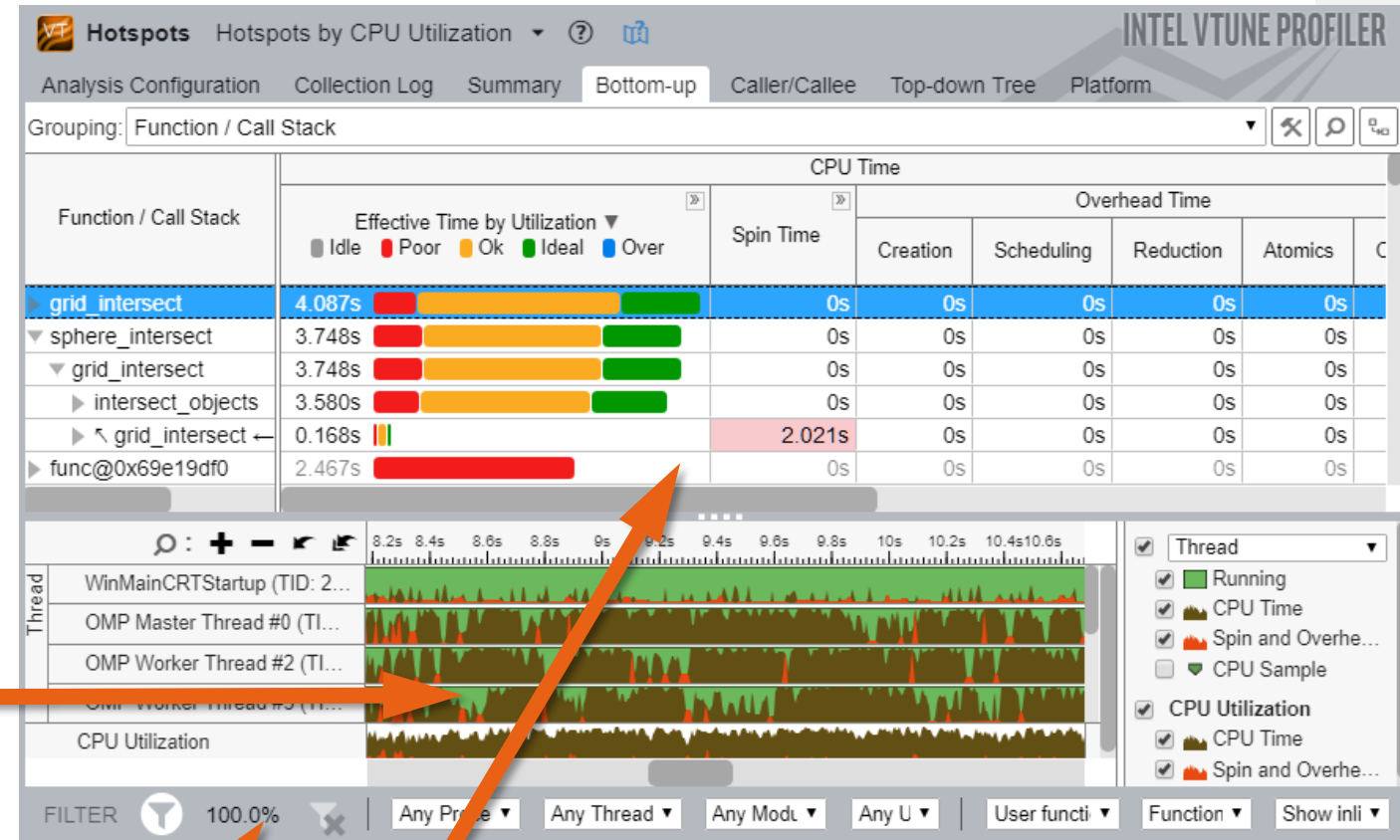
Function / Call Stack
Source Function / Function / Call Stack
Sync Object / Function / Call Stack
Sync Object / Thread / Function / Call Stack
... (Partial list shown)

Double Click Function
to View Source

Click [▶] for Call Stack

Filter by Timeline Selection
(or by Grid Selection)

Zoom In And Filter On Selection
Filter In by Selection
Remove All Filters



Filter by Process
& Other Controls

Tuning Opportunities Shown in Pink.
Hover for Tips

See Profile Data On Source / Asm

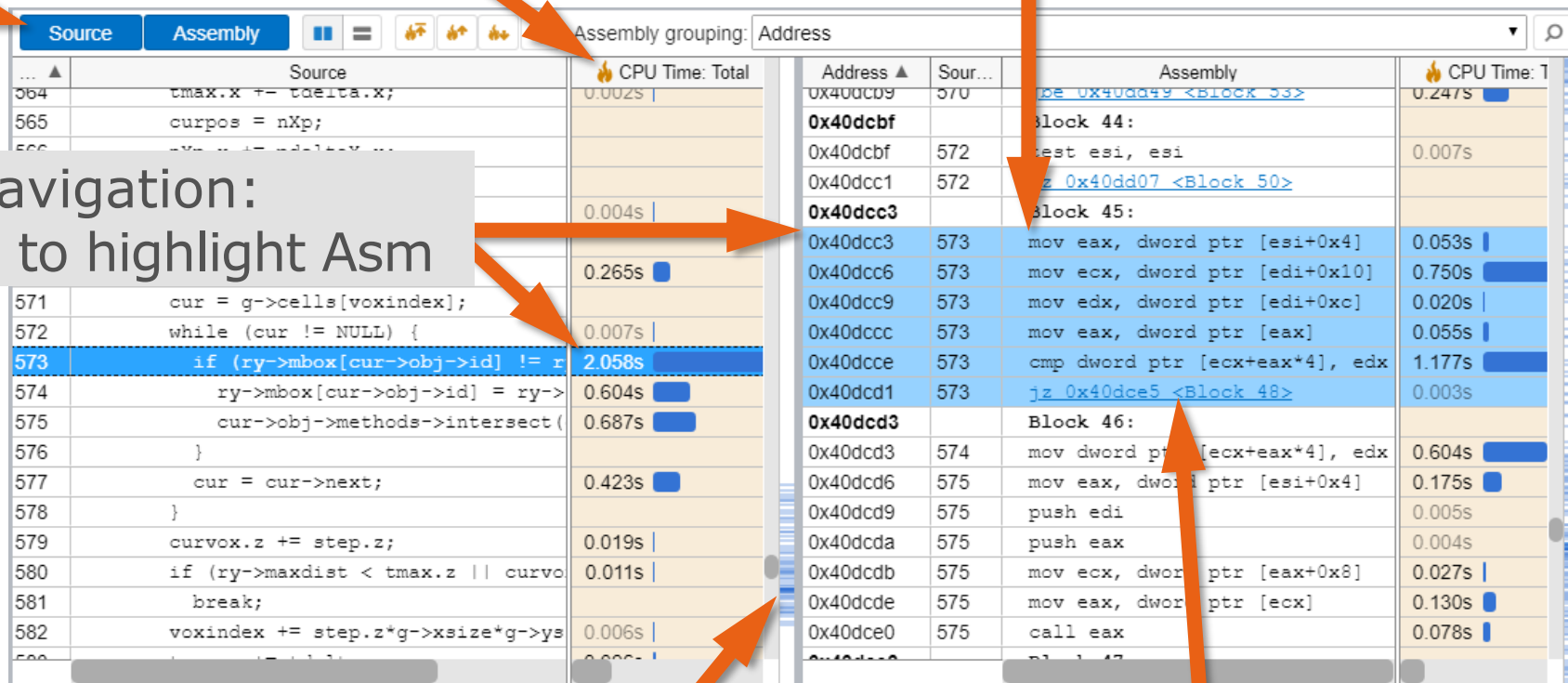
Double Click from Grid or Timeline

View Source / Asm or both

CPU Time

Right click for instruction reference manual

Quick Asm navigation:
Select source to highlight Asm



Scroll Bar "Heat Map" is an overview of hot spots

Click jump to scroll Asm

Bottom-up tab – most popular tab

- Grouping: different ordering of results –check out different choices
- Source view: double click on function or loop will open another window – source must be compiled with “-g”. Source code must be available
- Zoom and filter in timeline section. Grid will adapt your choice
- Filter process/user code/libraries/loops/system functions (e.g. libc)
- Values in pink: e.g. high overhead like barrier waiting or low cpu utilization.

How to start an analysis GUI vs CMD

- Start GUI by \$ `vtune-gui` on command line or by double clicking on Windows
- Click on “Configure Analysis ...”
- 3 Sections: WHERE,WHAT, HOW
- **WHERE**: local vs remote. We do local here
- **WHAT** : define your application with parameters and environment
- **HOW**: Analysis type like “Hotspots” or “APS” with additional parameters

WHERE

Local Host ▼

WHAT

 Launch Application ▾

Specify and configure your analysis target: an application or a script to execute. Follow [Prepare Application for Analysis](#) to compile your app for best analysis productivity.

Application:

C:\Users\hbockhor\OneDrive - Intel Corporation\Documents\Issues\2023\JIRA VASP-2861

Application parameters:

☒ Use application directory as working directory

Advanced

HOW

Hotspots ▼

Identify the most time consuming functions and drill down to see time spent on each line of source code. Focus optimization efforts on hot code for the greatest performance impact. [Learn more](#)

☒ User-Mode Sampling [?](#)

- Hardware Event-Based Sampling ?

☒ Show additional performance insights

Details

Collect CPU sampling data

With stacks

CPU sampling interval ms

10

Collect synchroniza

No

Collect signalling API data

No

Copy command line here

Command line

- Copy command line from GUI if you are not sure about the right configuration
- VTune has extensive help menu:
\$ vtune -help
\$ vtune -help collect
\$ vtune -help collect hotspots
- Check "Playbook.txt" for example command lines and additional flags

VTune and MPI

- To run VTune in an MPI job you may use the “-gtool” flag
- More convenient is the I_MPI_GTOOL environment variable. Example for HPC analysis:

```
$ export I_MPI_GTOOL= "vtune -c hpc-performance -r HPC:0"
```

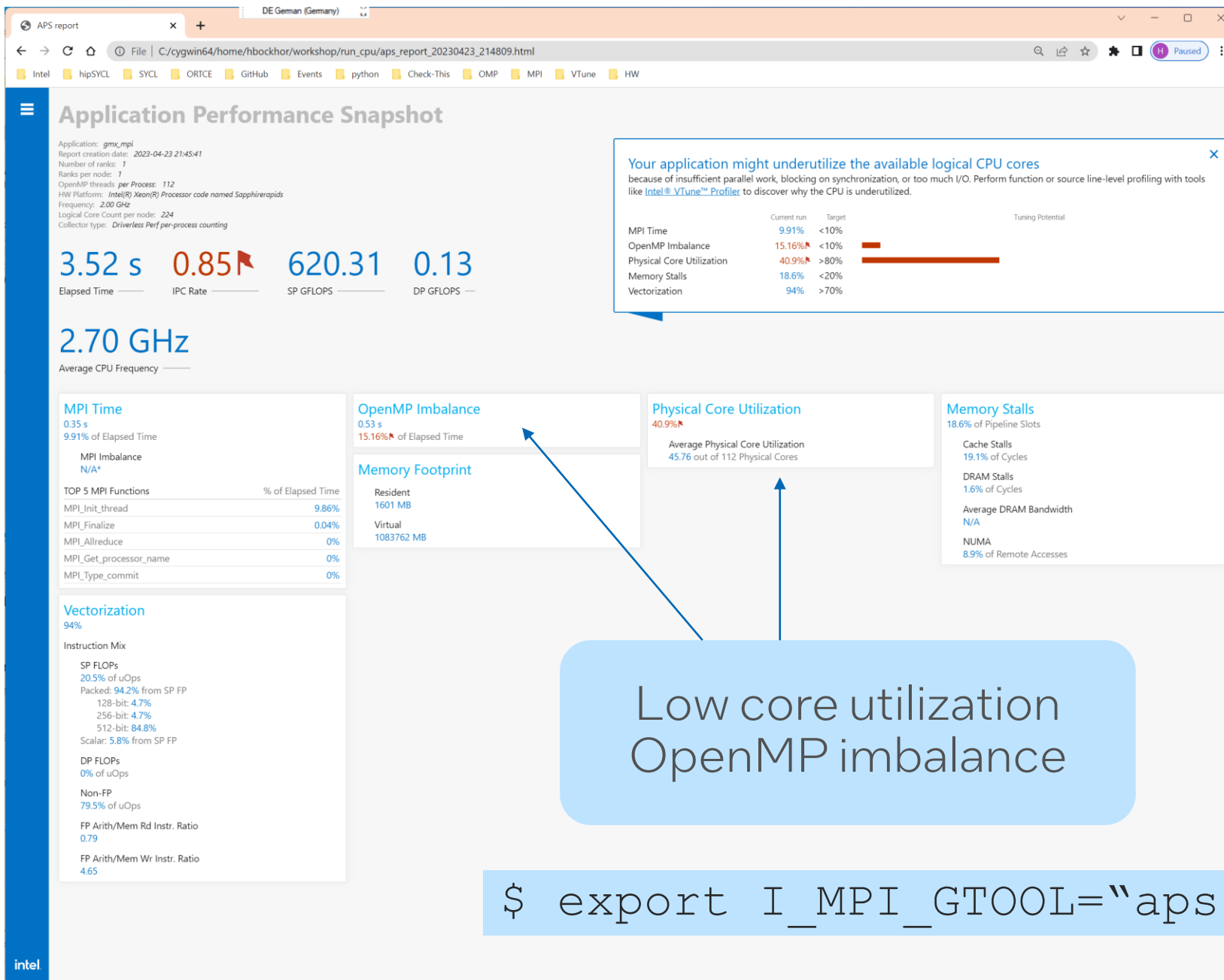
run your program, as usual, under MPI. The setting will collect data on rank #0. Use a list of ranks or :all for multi rank analysis.

- More information:

<https://www.intel.com/content/www/us/en/develop/documentation/mpi-developer-reference-linux/top/command-reference/mpiexec-hydra/gtool-options.html>

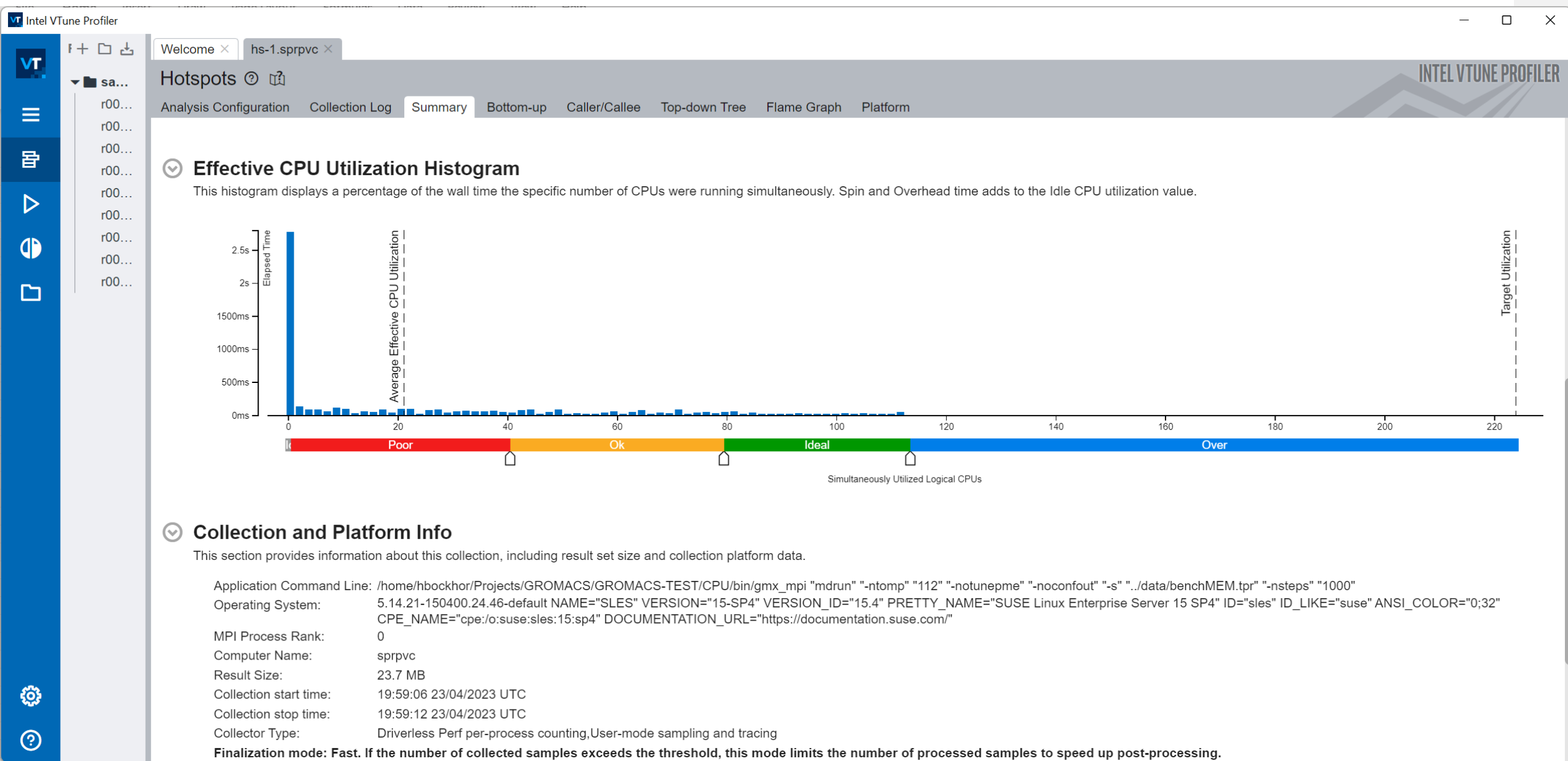
GROMACS description

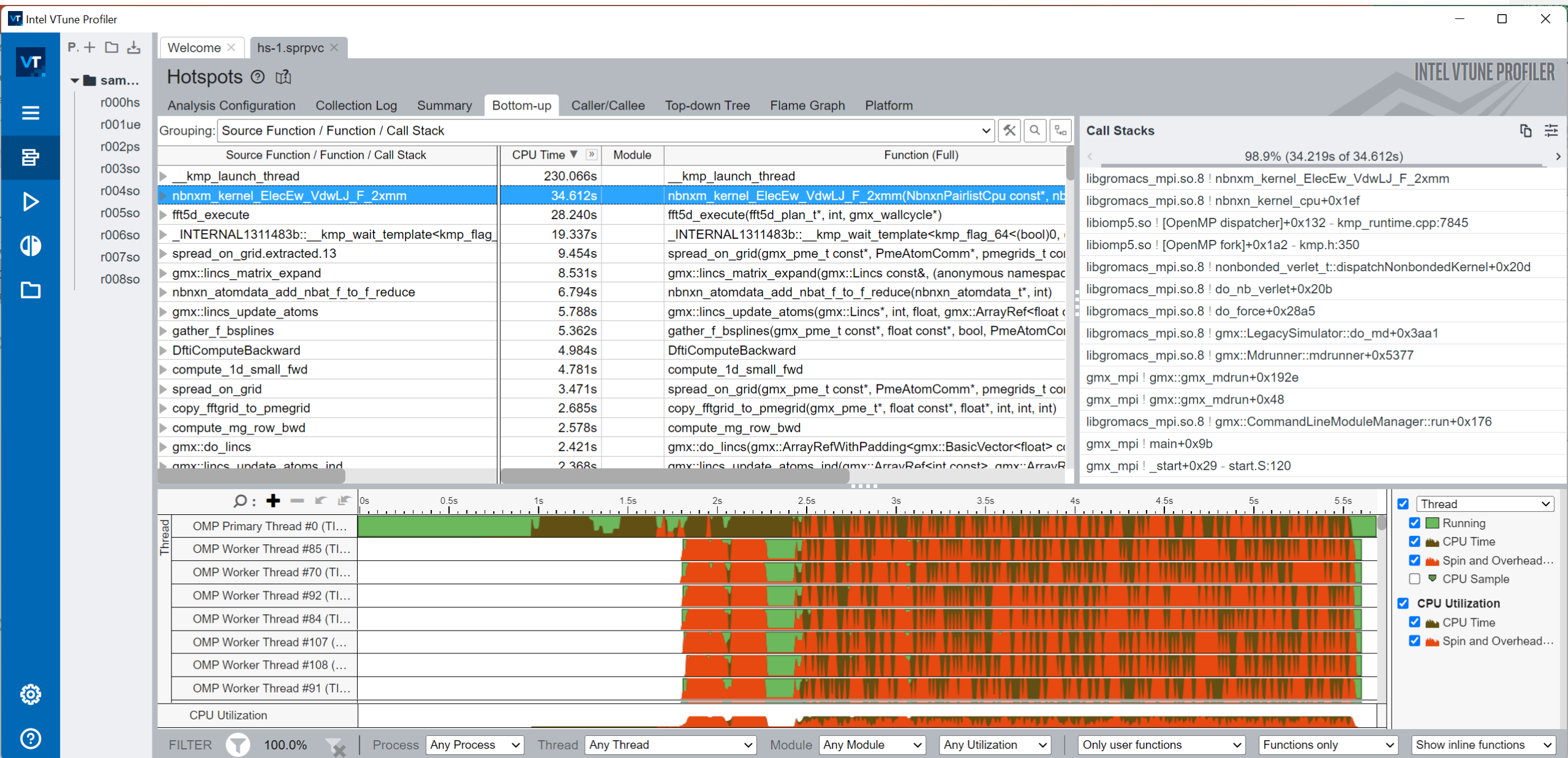
- Molecular Dynamics code with highly tuned CPU and GPU branch
- CPU branch uses AVX-512 intrinsics – cmake configuration
- Classical intel compilers (icc, icpc) were deprecated
- Calculation of forces has 2 large parts: direct calculations and large distance forces using FFTs (Particle Mesh Ewald PME)
- Everything is tuned but (intended) naïve execution shows some room for improvement

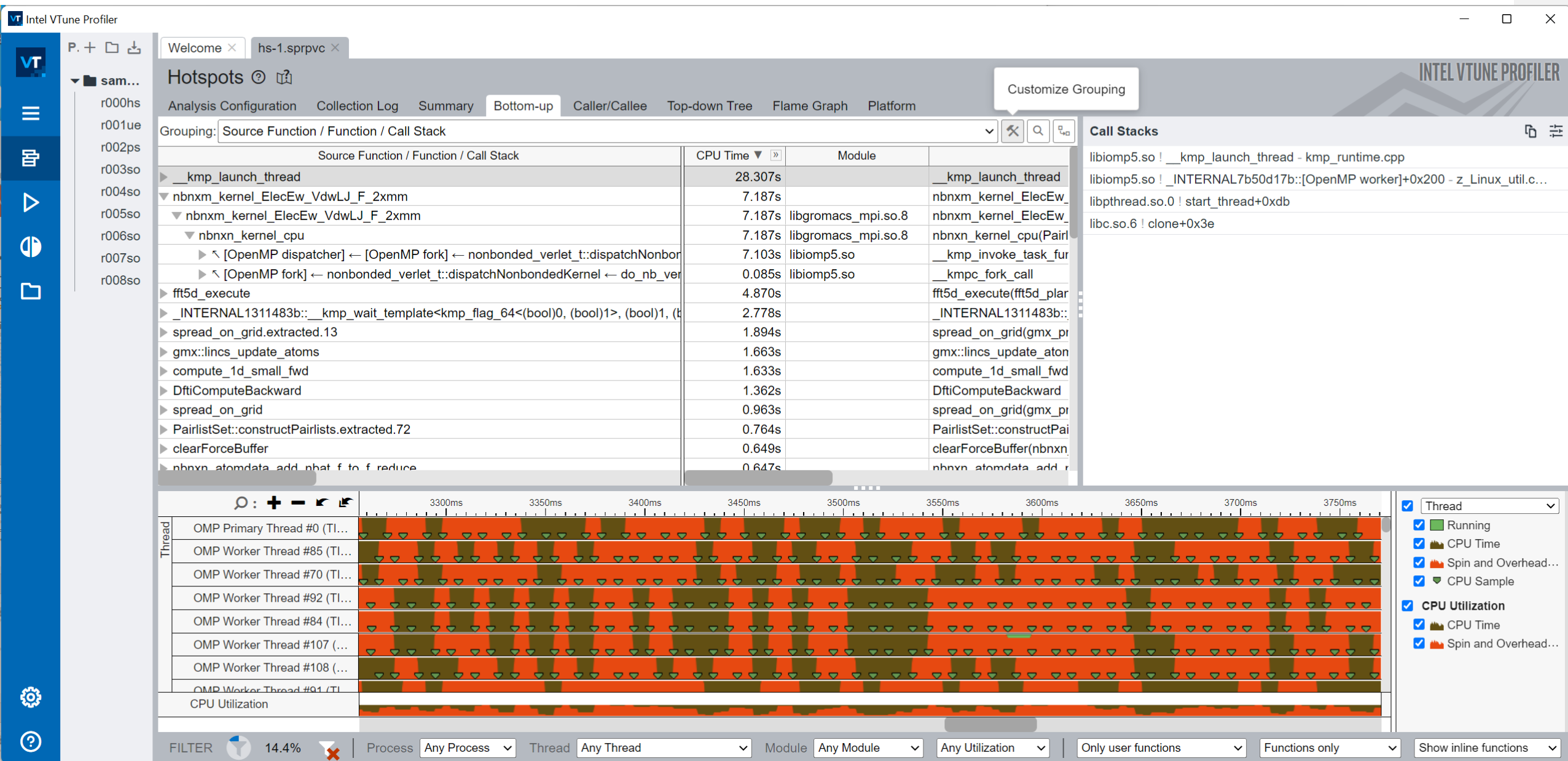


APS information

- Low logical core utilization : only single thread per core used
- High OMP imbalance: this can be due to issues with the algorithm or a bad configuration of the environment
- Next step(s): Use VTune with hotspot analysis
- hpc-performance might be better choice, but this comes later ...
- Check command line in the playbook







Notices & Disclaimers

Intel technologies may require enabled hardware, software or service activation. Learn more at intel.com or from the OEM or retailer.

Your costs and results may vary.

Intel does not control or audit third-party data. You should consult other sources to evaluate accuracy.

Optimization Notice: Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice. Notice Revision #20110804. <https://software.intel.com/en-us/articles/optimization-notice>

Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors.

Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. See backup for configuration details. For more complete information about performance and benchmark results, visit www.intel.com/benchmarks.

Performance results are based on testing as of dates shown in configurations and may not reflect all publicly available updates. See configuration disclosure for details. No product or component can be absolutely secure.

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

© Intel Corporation. Intel, the Intel logo, and other Intel marks are trademarks of Intel Corporation or its subsidiaries. Other names and brands may be claimed as the property of others.

