Promoting Efficiency in European HPC

Sally Bridgwater, Mike Dewar, Nick Dingle, Numerical Algorithms Group Ltd, Oxford, UK
Judit Gimenez, Jesus Labarta, Claudia Rosas, Barcelona Supercomputing Center, Spain
Aamer Shah, RWTH Aachen, Germany

Aims of POP

The *Performance Optimisation and Productivity Centre of Excellence in Computing Applications* (POP) provides independent performance assessments of academic and industrial HPC codes.

POP enables code owners to improve the efficiency of their codes and thus make better use of HPC resources and analyse larger or more detailed problems.

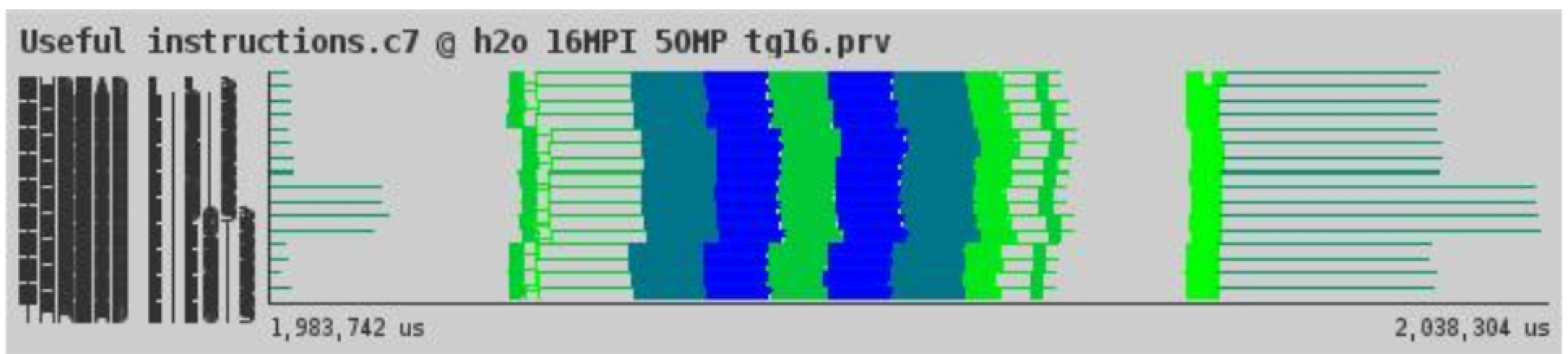
How to Apply

The POP website **`https://pop-coe.eu`** provides details of the services available and how to apply.

The service is free to organisations based in the EU.

Case Study: Quantum Espresso

Quantum Espresso is an integrated suite of Open Source codes for nanoscale electronic structure calculations and materials modelling.



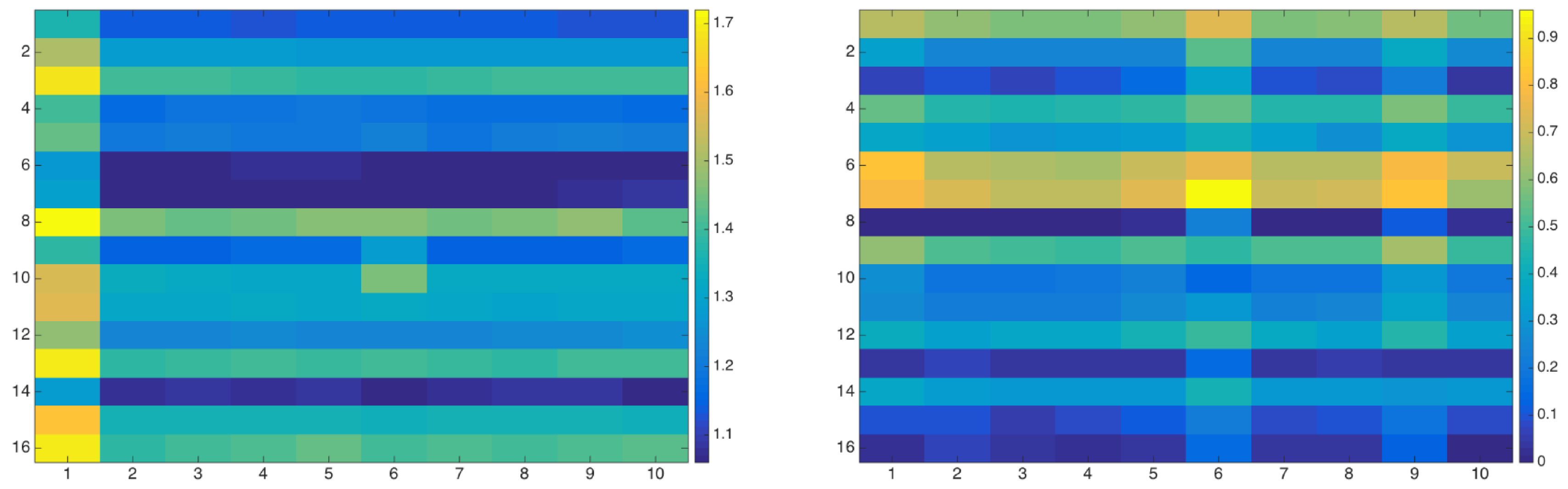
POP investigated the effectiveness of its hybrid MPI+OpenMP architecture and found that for a significant portion of time only one OpenMP thread per MPI process was doing useful computation.

This explains the observation that running with five OpenMP threads was less than twice as fast as running with one. Refactoring the code to better exploit all OpenMP threads would improve its scalability.

This work was done in collaboration with the *Materials design At the eXascale* (MaX) CoE.

Case Study: DROPS

DROPS is a tool for simulating two-phase flows. POP identified a computational load imbalance in the matrix setup stage that increased waiting times in MPI collective operations.

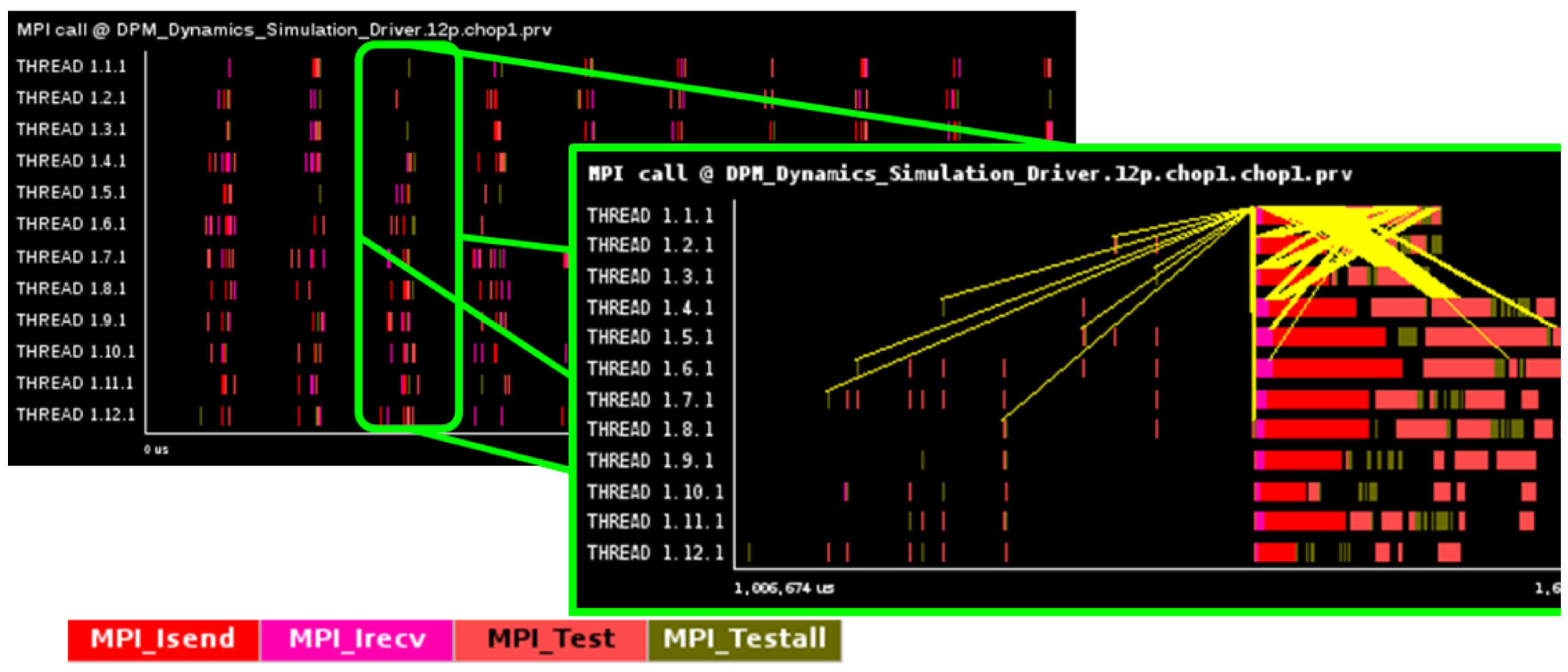


This is visualized in the above heat maps: the left shows the computation time in seconds per MPI rank, the right the idle time.

POP identified the cause of this load imbalance and, as it severely affected the performance of the code, recommended that it should be a priority to rectify it.

Case Study: DPM

Discrete Particle Method (DPM) is a numerical simulation tool for studying the motion and chemical conversion of particulate material in furnaces.



POP identified issues with end-point contention due to sending MPI messages in increasing-rank order.

This contention prevented the code from scaling beyond 20 cores. It can be avoided if different MPI ranks send their messages in different orders.

https://pop-coe.eu

This project received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 676553