The POP Centre of Excellence in HPC

Fouzhan Hosseini, The Numerical Algorithms Group (NAG)

Fouzhan.hosseini@nag.co.uk, Nov 2020

EU H2020 Centre of Excellence (CoE)

1 December 2018 – 30 November 2021

Grant Agreement No 824080

Performance Optimisation and Productivity

A Centre of Excellence in HPC

- Promotes best practices in parallel programming
  - Improving Parallel Software can add a lot of value: Reduced expenditure, faster results, novel solutions
  - **The POP Methodology -** a systematic approach to performance optimization building a quantitative picture of application behavior

- Free services for all EU academic and industrial codes and users
  - Suggestions on improving code performance, described in a ***Performance Assessment***
  - Practical help with code refactoring through a ***Proof of Concept***

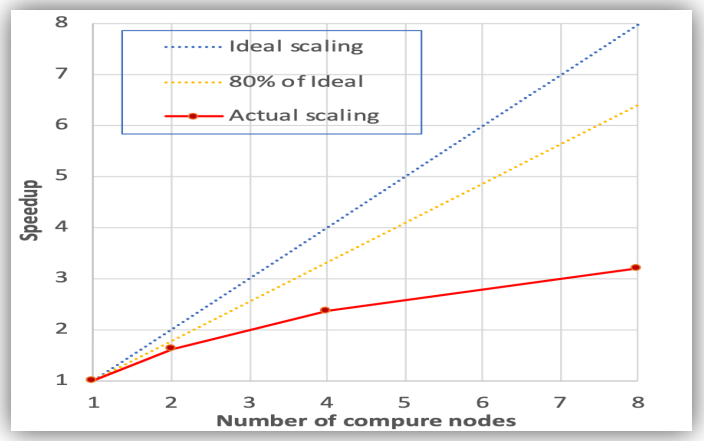
- A Team with
  - Excellence in performance tools and tuning
  - Excellence in programming models and practices
  - R & D background in real academic and industrial use cases

For further information, visit: https://www.pop-coe.eu pop@bsc.es @POP_HPC youtube.com/POPHPC

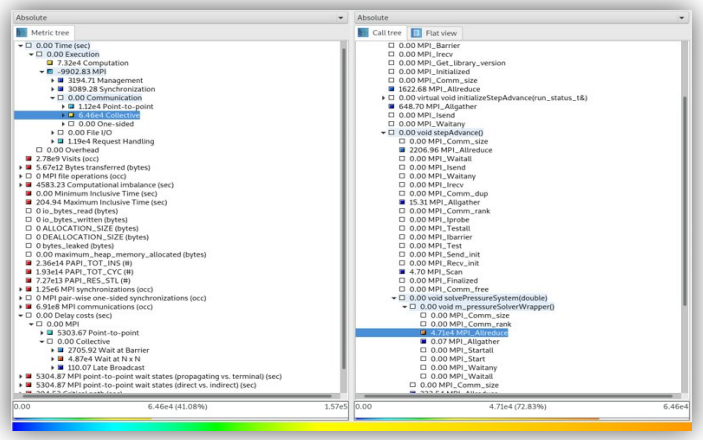
Parallel Performance is hard to understand

How do we measure the performance of our parallel programs?

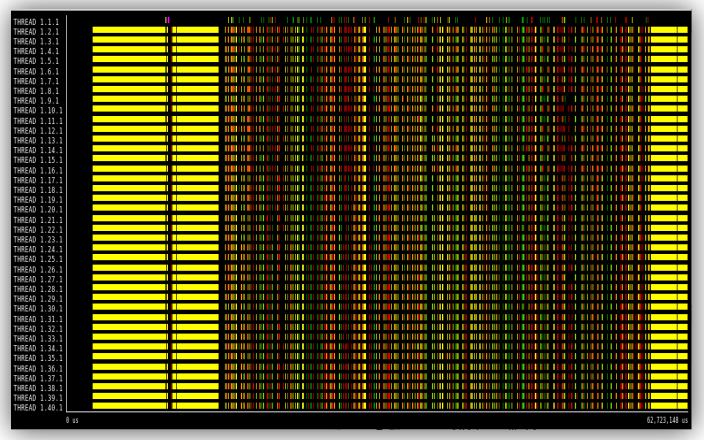
- Traditional speed-up and efficiency plots?
- Profiling & tracing with performance tools?
  - Tracing is powerful, but potentially generates overwhelming amount of data



Speedup plot



Cube, perf. metrics per routines/call stack, data collected by Scalasca/Score-P



Paraver, timeline view of program execution, data collected by Extrae

Difficult to know where to start and what to look for

Main Problem: Lack of quantitative understanding of the actual behavior of a parallel application

A Solution:

The POP Metrics

Simple but extremely powerful idea

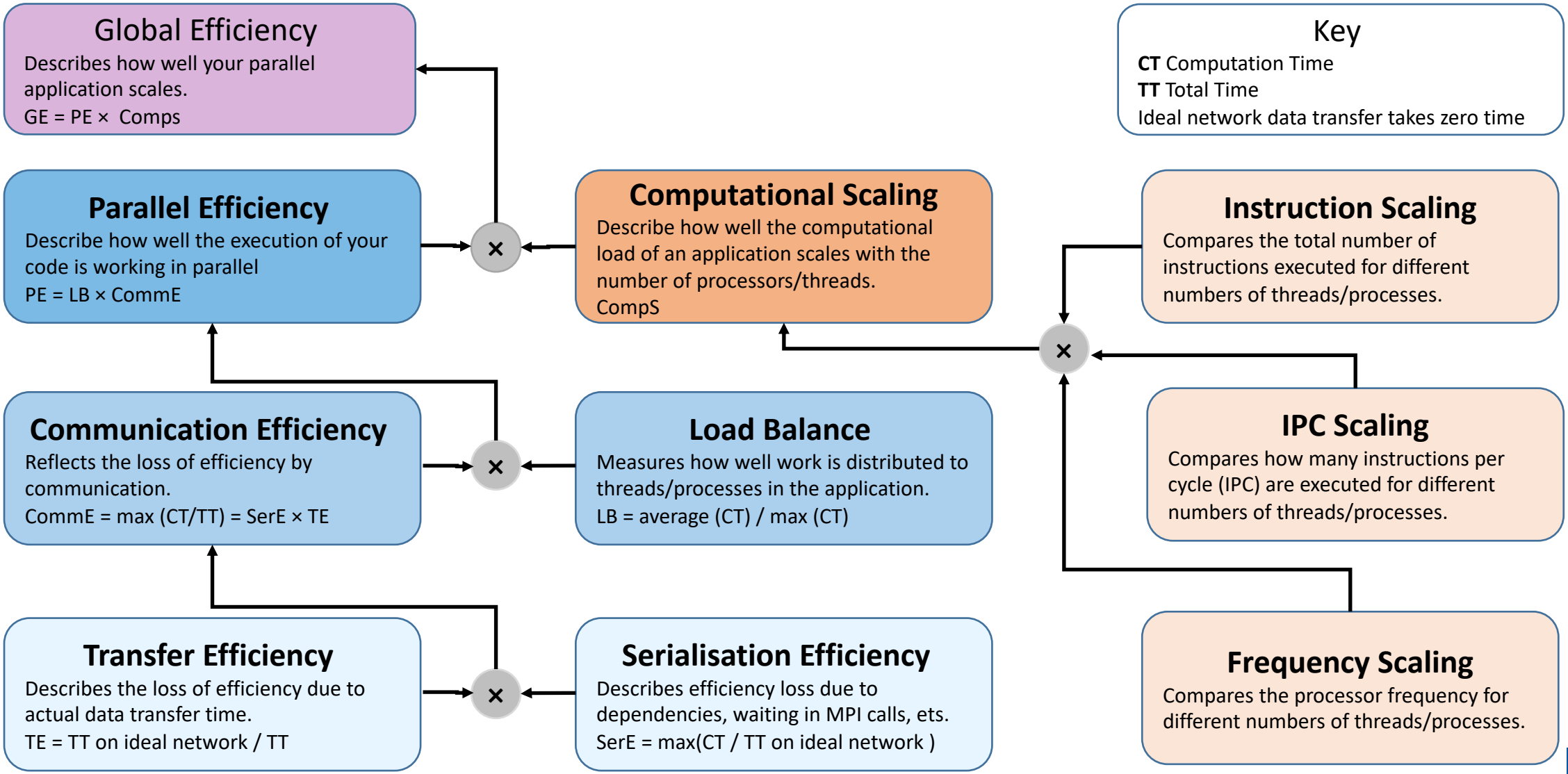
- Devise a simple set of performance metrics using values easily obtained from the trace data
- Where low values indicate **specific** causes of poor parallel performance

These metrics then are used to understand

- What are the causes of poor performance
- What to look for in the trace data

- Besides, the metrics provide a common ground for discussing performance issues
  - Between developers, users and analysts

POP MPI Parallel Efficiency Metrics



For more details visit https://pop-coe.eu

POP Metrics Example

Number of cores	48	96	192	384	768
Global Efficiency	0.93	0.94	0.93	0.84	0.76
↳ Parallel Efficiency	0.93	0.91	0.87	0.77	0.68
↳ Load balance	0.99	0.98	0.98	0.97	0.95
↳ Communication Efficiency	0.94	0.92	0.89	0.79	0.72
↳ Serialisation	0.95	0.94	0.92	0.85	0.81
↳ Transfer efficiency	0.99	0.99	0.97	0.94	0.89
↳ Computational Scaling	1.00	1.03	1.07	1.09	1.12
↳ Instruction Scaling	1.00	0.99	0.97	0.95	0.92
↳ IPC Scaling	1.00	1.05	1.10	1.18	1.27
↳ Frequency Scaling	1.00	1.00	1.00	0.98	0.96

- We immediately see that **Serialisation** is the main factor that limits the scalability
- Efficiency values are between 0 to 1, and
  - metric values above 0.8 represent acceptable performance

Performance Optimisation and Productivity

A Centre of Excellence in HPC

Free training materials
Regular Webinars
Service request form

Contact:

- https://www.pop-coe.eu
- pop@bsc.es
- @POP_HPC
- youtube.com/POPHPC

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 676553 and 824080.