Benchmark description

June 23, 2020

Throughput and strong-scaling benchmarks

This benchmark suite is composed of application benchmarks and synthetic benchmarks. The list of application benchmarks contains research software packages heavily used in HPC, i.e. Gromacs, Nek5000, VASP, PowerFLOW, Converge, PyFR and CP2K-Quickstep.

These benchmarks fall into two classes depending on how they should be run and how they will be evaluated:

- Throughput benchmarks
- Strong-scaling benchmarks

Throughput benchmarks

- Use case: regular HPC research where a certain number of jobs need to be completed in order to finish a research project, without wasting core hours.
- Figure of merit (result): Number of jobs that run on the system in one day.
- Formula:

$$n_{\text{jobs}}(b) = \frac{t_{\text{day}}}{t(b)} \cdot \frac{N_{\text{tot}}}{N(b)} \tag{1}$$

  where:

  - t_{day} = time in a day (86400 s)
  - t(b) = average time in seconds to run one job for benchmark b
  - N_{tot} = total number of nodes tendered
  - $N(b)$ = number of nodes used to run benchmark b
  - $n_{\text{jobs}}(b)$ = number of jobs that can be run in a day for benchmark b. This result must be reported for each throughput benchmark in the benchmark matrix spreadsheet.

- All throughput benchmarks have a **minimum performance** to ensure that more than one node needs to be used.

The throughput tests represent a trade-off between maximizing the number of nodes in the system, minimizing the time it takes to run one job and minimize the number of nodes used for each job. Each throughput benchmark case has a specified minimum performance which most likely requires it to be run on more than one node.

Strong scaling benchmarks

- Use case: Researcher needs to finish a massively parallel job in the shortest possible time. The researcher has access to the entire system.
- Figure of merit (result): Number of jobs (each job solves the problem in the shortest possible time) that run on the system in one day.

- Formula:

$$n_{\text{jobs}}(b) = \frac{t_{\text{day}}}{t(b)} \tag{2}$$

  where:

  - t_{day} = time in a day (86400 s)
  - $t(b)$ = average time in seconds to run one job in the shortest possible time for benchmark b
  - $n_{\text{jobs}}(b)$ = number of jobs that can be run in one day for benchmark b. This result must be reported for each strong-scaling benchmark in the benchmark matrix spreadsheet.

In the strong scaling benchmarks the goal is simply to minimize the runtime of the benchmark cases by using as many nodes/cores/devices as required to maximize performance.

The calculation of the throughput and strong-scaling figures of merit for each benchmark case is clarified in the corresponding benchmark documentation.