Parallel Programming

Timings

Prof. Paolo Bientinesi

HPAC, RWTH Aachen
pauldj@aices.rwth-aachen.de

WS17/18

Time

- **Wall time** or “wall-clock time” :
  real time between the beginning and the end of a computation

- $T_p(n)$:= Wall time to solve a problem of size n using p processes

  $T_p(n) = t_1 - t_0$
  t_0 : time when the first process starts its execution,
  t_1 : time when the last process completes its execution

- **CPU-time** or “core time” :
  cumulative time spent by all processes in a computation

Code

- `time0.c`
  Cholesky factorization.
  No timings. Only correctness.

- `time1.c`
  Timings through `clock()`.
  Multithreading (via LAPACK/BLAS). CPU-time.

- `time2a.c`
  Cycle accurate timer.
  Cycles, frequency. Wall time vs. CPU-time.

- `time2b.c`
  Performance (# ops/sec), efficiency.

Performance

- **Performance**: Number of floating point operations per second performed while solving a given problem.

- **Theoretical Peak Performance** (TPP): In ideal conditions, the highest number of floating point operations that a processor can perform in one second.

- **Peak Performance** “Practical peak performance”: The performance attained by highly tuned matrix-matrix multiplication kernels (DGEMM). For instance, MKL and OpenBLAS.

- **Efficiency**: The ratio between the performance attained while solving a given problem and the TPP (or the PPP).