HPL benchmark

Uroš Lotrič, Davor Sluga

April 2024

1 The HPL Benchmark

The HPL benchmark, short for High-Performance Linpack, is a widely used benchmarking tool in high-performance computing (HPC). Its primary purpose is to evaluate the floating-point performance of computer systems, particularly supercomputers and clusters. HPL essentially solves a dense system of linear equations using LU decomposition with partial pivoting, utilizing the LINPACK numerical library. HPL involves intense matrix computations, including matrix factorizations, multiplications, and other linear algebra operations. It is designed to take full advantage of parallel processing capabilities offered by modern HPC architectures. It is based on OpenMP and MPI libraries, so it can utilize multiple processors, nodes, or cores efficiently, making it suitable for evaluating the performance of parallel computing systems. Consequently, it has become a standard benchmark in the HPC community, providing a common metric for comparing the performance of different computing systems. HPL is often one of the first programs run on large computer installations to produce a result that can be submitted to TOP500 to compare it to other systems worldwide. Overall, the HPL benchmark plays a crucial role in the development, testing, and optimization of high-performance computing systems.

2 Assignment

Your task is to evaluate the performance of a single CPU node on the Arnes cluster using the HPL benchmark. The CPU present on Arnes computing nodes is the 64 core AMD Epyc 7702P, which is based on the 2nd generation of AMD EPYC data centre-oriented CPUs. For this CPU, the company states it can achieve $R_{peak} = 2048$ GFLOPS (billion floating point operations per second). This number is calculated the following way: $R_{peak} = f_{cpu} \times C \times DFP$, where f_{cpu} is the CPU frequency (2.0 GHz), C is the number of cores (64), and DFP is the number of floating point operations which can complete every CPU cycle on each CPU (16). Of course, in practical applications, such performance is unobtainable. but your task is to see how close to this performance one can get using the HPL benchmark.

We will use the prebuilt AMD binary of the HPL benchmark, which is optimized for the AMD Zen microarchitecture. You can download the benchmark here. Additional information on how to run the benchmark can be found here. Carefully read the README.md and TUNING.md files, which are included with the HPL application. Inside, you will find information on configuring the benchmark to run as fast as possible. Some additional information on using the benchmark can be found here.

Notes:

- The benchmark is configured through the HPL.dat configuration file, where you define the size of the matrices, etc.
- The benchmark performs best on very large matrices. You must find the maximum matrix size to fit into the compute node memory.
- When running the benchmark, allocate the whole node (128 cores) to obtain the best performance. When troubleshooting and configuring the benchmark, use fewer cores so that you leave some CPU cores for other users.
- The benchmark can be run using OpenMP, MPI, or both.
- The distribution of HPL threads/processes across the CPU cores is important. The distribution can be configured using OpenMP environment variables or switches of the mpirun command.
- Set the time and memory limits correctly for your job. These are set using sbatch switches *--time* and *--mem*.

Tasks:

- Configure and run the benchmark on a single CPU node of the Arnes cluster. Try to achieve as high GFLOPS as possible.
- Run the benchmark using only OpenMP and using only MPI. Is there a difference in performance?
- Write a short report (1-2 pages) summarizing the obtained performance. Include the contents of the configuration and job description files in the report (HPL.dat and sbatch script) which yielded you the best score.
- In the report, explain the settings you used to obtain the highest performance.
- Hand in scripts and the report to ucilnica through the appropriate form by the specified deadline (**12. 5. 2024**). There will be no defence of this assignment during labs.