Benchmarking with HPL and Intel Optimized LINPACK Benchmark

Simone Amadio

The aim of this exercise is to run both the High Performance LINPACK and the Intel Optimized LINPACK Benchmark, in order to test the Sustained Peak Performance of a node in the Ulysses cluster.

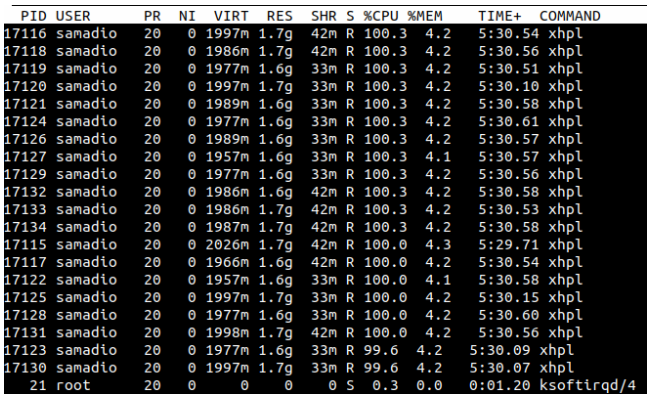
1 HPL library

1.1 One node, 20 processors

In the input file of the HPL library, named HPL.dat, there are a few parameters which in principle should affect the Sustained Peak Performance:

- Ns: (Linear) Dimension of the matrix. It should take $\sim$ 75% of memory. Calculated through the site http://www.advancedclustering.com/act-kb/tune-hpl-dat-file/
- Nbs: Number of blocks. Values suggested from the provider is 192 or 256.
- Ps: rows of process grid ratio
- Qs: columns of process grid ratio. Values of P and Q are suggested to be as equal as possible

I run the HPL benchamrk using mpirun with 20 processes and I connected through ssh to the single node in which the benchamrk was running. I used the command top in order to see the hardware usage and this showed that every CPU was completely exploited.
The program was run many times in order to get an average, but due to the strcture of the queue system the program has been executed each time in a different node. After comparing the results of the same program, we get the following results.
The blue line represent the Theoretical Peak Performance(TTP) of a node in the Ulysses cluster, which is 448 GFlops. The graph shows clearly the benchmark achieves more than 90% of the TTP, with a peak of nearly 95%.
Despite this excellent results, the oscillations of the values obtained are nearly 5%, which means that varying a little the input parameter in the file HPL.dat would not give any observable result.
For this reason I decided not to collect any more data



```
  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
17116 samadio   20   0 1997m 1.7g  42m R 100.3  4.2   5:30.54 xhpl
17118 samadio   20   0 1986m 1.7g  42m R 100.3  4.2   5:30.56 xhpl
17119 samadio   20   0 1977m 1.6g  33m R 100.3  4.2   5:30.51 xhpl
17120 samadio   20   0 1997m 1.7g  33m R 100.3  4.2   5:30.10 xhpl
17121 samadio   20   0 1989m 1.6g  33m R 100.3  4.2   5:30.58 xhpl
17124 samadio   20   0 1977m 1.6g  33m R 100.3  4.2   5:30.61 xhpl
17126 samadio   20   0 1989m 1.6g  33m R 100.3  4.2   5:30.57 xhpl
17127 samadio   20   0 1957m 1.6g  33m R 100.3  4.1   5:30.57 xhpl
17129 samadio   20   0 1977m 1.6g  33m R 100.3  4.2   5:30.56 xhpl
17132 samadio   20   0 1986m 1.6g  42m R 100.3  4.2   5:30.58 xhpl
17133 samadio   20   0 1986m 1.7g  42m R 100.3  4.2   5:30.53 xhpl
17134 samadio   20   0 1987m 1.7g  42m R 100.3  4.2   5:30.58 xhpl
17115 samadio   20   0 2026m 1.7g  42m R 100.0  4.3   5:29.71 xhpl
17117 samadio   20   0 1966m 1.6g  42m R 100.0  4.2   5:30.54 xhpl
17122 samadio   20   0 1957m 1.6g  33m R 100.0  4.1   5:30.58 xhpl
17125 samadio   20   0 1997m 1.7g  33m R 100.0  4.2   5:30.15 xhpl
17128 samadio   20   0 1977m 1.6g  33m R 100.0  4.2   5:30.60 xhpl
17131 samadio   20   0 1998m 1.7g  42m R 100.0  4.2   5:30.56 xhpl
17123 samadio   20   0 1977m 1.6g  33m R 99.6  4.2   5:30.09 xhpl
17130 samadio   20   0 1997m 1.7g  33m R 99.6  4.2   5:30.07 xhpl
   21 root      20   0     0    0    0 S  0.3  0.0   0:01.20 ksoftirqd/4
```

Top output for the HPL benchmark

1.2 Two nodes, 40 processors

The same process was repeated

2 Appendix: commands and code

2.1 HPL library

In order to download and install the library, I run the following commands:

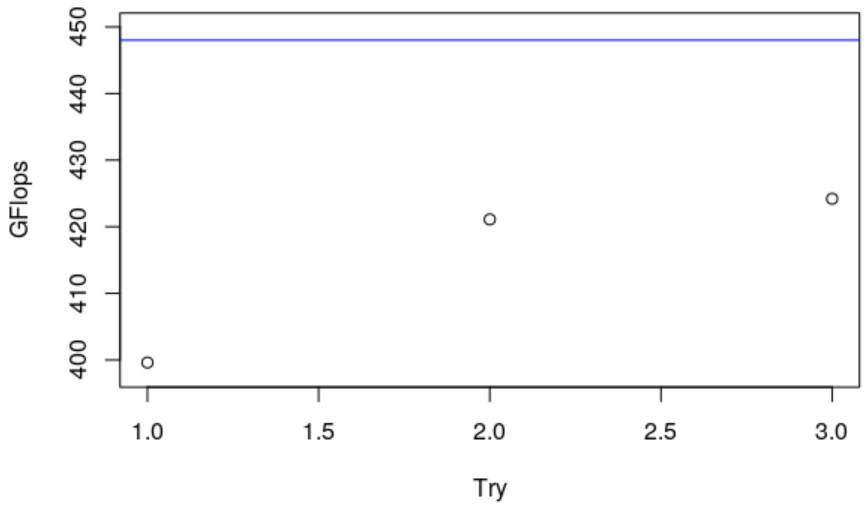
```
module load mkl
module load openmpi/1.8.3/intel/14.0
wget http://www.netlib.org/benchmark/hpl/hpl-2.2.tar.gz
tar -xvzf hpl-2.2.tar.gz
cd hpl-2.2
cd setup/
cp Make.Linux_Intel64 ../.
cd ..
mv Make.Linux_Intel64 Make.mkl
```

Then I proceeded to edit some lines in the Make.mkl file:

```
ARCH = mkl
TOPdir = $(HOME)/hpc/ex6/hpl-2.2
# MPdir        = /opt/intel/mpi/4.1.0
# MPinc        = -I$(MPdir)/include64
# MPlib        = $(MPdir)/lib64/libmpi.a
LAdir        = $(MKLROOT)
```

Finally, I run the Makefile in order to create the executable ./xhpl

```
make arch=mkl
```



Results of HPL benchmark in different nodes

2.2 Intel Optimized LINPACK BENCHMARK

```
module load mkl
cd $MKLROOT/ benchmarks/linpack/
cp /u/shared/programs/x86_64/mkl/11.1.3/composer_xe_2013_sp1.3.174/
        mkl/benchmarks/linpack/xlinpack_xeon64 .
cp /u/shared/programs/x86_64/mkl/11.1.3/composer_xe_2013_sp1.3.174/
        mkl/benchmarks/linpack/lininput_xeon64 .
```

2.3 bash script

```
## a small script to execute my benchamrk

/bin/hostname #name of the node which is executing my job
#enter in the right position
cd hpc/ex6/hpl-2.2/bin/mkl/nthreads

#load modules needed
module load mkl
module load openmpi/1.8.3/intel/14.0

#run HPL Benchmark

mpirun -np 20 ./xhpl
#now the intel
export OMP_NUM_THREADS=5
./xlinpack_xeon64 lininput_xeon64
```

```
exit
```