Exercise 3: Approximate Pi with MPI vs OpenMP

Gabriele Sarti

May 13, 2019

Abstract

In the third exercise for the course of Parallel computing, our aim is to compare the times of computing a Pi approximation with OpenMP and MPI.

1 Procedure and Results

Table 1 presents the times of execution for OpenMP Pi approximation using a size of $n = 1000000000$ on mpicc compiler with an optimization level of -O3.

Table 1: Results of Exercise 1 execution

# threads	OMP Reduction time (s)
1	5.29
2	2.67
4	1.37
8	0.76
16	0.40
20	0.32

Table 2 presents the times of execution for MPI Pi approximation using a size of $n = 1000000000$ on mpicc compiler with an optimization level of -O3.

From the results we may observe that multiprocessing seems to be more efficient time-wise to perform the task of approximating the value of Pi.

Table 2: Results of Exercise 3 execution

# processes	MPI Reduction time (s)
2	1.00
4	0.50
8	0.27
12	0.19
16	0.14
20	0.11
24	0.08
28	0.08
32	0.07
36	0.06
40	0.06

2 Reproducibility

In order to obtain results that are similar to those listed above, simply clone the Github repository [1] in the personal Ulysses folder and run the following commands from your main folder:

```
# For reproduce OpenMP results
qsub -q reserved3 -l nodes=1:ppn=20 parallel-computing/Assignments/ex01/ex1.sh
# For MPI results
qsub -l nodes=2:ppn=20 parallel-computing/Assignments/ex03/ex3.sh
```

The result will be contained in the files ex1.sh.o* and in the file times.txt which will be created in the Assignments folder.

References

[1] https://github.com/gsarti/parallel-computing