Introduction to Supercomputing

TMA4280 · Problem set 4

Note that

- this problem set is mandatory;
- you can work on it in groups with *up to* 3 members;
- you should write a report describing your solution, as continuous text;
- the source code should be handed in together with the report (and not as part of it);
- please make sure that you have answered all the questions;
- the due date is Friday, February 26, 2015;
- the report will count 10% towards the final grade.

We consider a vector $v \in \mathbb{R}^n$ where the vector elements are defined as

$$v_i = \frac{1}{i^2}, \qquad i = 1, \ldots, n.$$

We are interested in computing the sum of all vector elements numerically, i.e. we want to find S_n, where

$$S_n = \sum_{i=1}^{n} v_i.$$

Note that the limit is

$$S = \lim_{n \to \infty} S_n = \frac{\pi^2}{6}.$$

Exercise 1. Write a program, in either C or Fortran, that

- generates the vector v;
- computes the sum S_n in double precision on a single process;
- computes the error $|S - S_n|$ for $n = 2^k$ with $k = 3, 4, \ldots, 14$;
- prints out the error $|S - S_n|$ in double precision.

Exercise 2. Make the necessary changes needed to use shared memory parallelization with OpenMP.

Exercise 3. Write a program that computes the sum S_n using a distributed memory model (MPI) on P processors, where P is a power of two.

The program should work as follows. Only process zero (the root) should be responsible for generating the vector elements. The root should partition and distribute the elements evenly among all the processes. Each process should sum up its own part. At the end, all the partial sums should be added together and made available on the root for printing. Report the error $|S - S_n|$ in double precision for different values of n.

Exercise 4. Confirm that your program also works when using OpenMP and MPI in combination.

Exercise 5. Which MPI calls were convenient and/or necessary to use?

Exercise 6. Compare the errors from the single-process program and the multi-process program for $P = 2$ and $P = 8$. Should the answer be the same in all cases? Exactly, or approximately?

Exercise 7. Compare the memory requirement per process for the single-process program and the multi-process program when $n \gg 1$.

Exercise 8. How many floating point operations are needed to generate the vector v? How many are needed to compute S_n? Is the multi-process program load balanced?

Exercise 9. Do you consider parallel processing attractive for solving this problem?