TMA4280: Introduction to Supercomputing

Problem set 4

Spring 2013

NOTE THAT:

- this problem set is mandatory;
- you can work on this problem alone or with one other student;
- a report describing the solution should be written (a continuous text is expected);
- please make sure that you have answered all the questions;
- the source code should be delivered together with your report;
- the due date is Monday, February 25, 2011;
- the report will count 10% towards the final grade.

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

$$v(i) = \frac{1}{i^2}, \qquad i = 1, \ldots, n. \tag{1}$$

We want to compute the sum of all the vector elements numerically, i.e., we want to find S_n where

$$S_n = \sum_{i=1}^{n} v(i). \tag{2}$$

Note that the limit $S = \lim_{n\to\infty} S_n = \pi^2/6$; e.g., see Rottman.

1. Write a program (either in C or in FORTRAN) which will do the following:
   - generate the vector $\underline{v}$;
   - compute the sum S_n in double precision on a single processor;
   - compute the difference $S - S_n$ for $n = 2^k$, with $k = 4, \ldots, 14$;
   - print out the difference $S - S_n$ in double precision.
2. Write a parallel version of the program suitable for shared memory computers.
3. Write a program to compute the sum S_n using P processors where P is a power of 2, suitable for distributed memory computers. Each processor should be responsible for generating n/P vector elements and summing up these vector elements. At the end, all the partial sums should be added together and collected on processor 0 for printout. Report the difference $S - S_n$ in double precision for different values of n.
4. Which MPI calls are convenient/necessary to use?
5. Compare the difference $S - S_n$ from the single-processor program and the multi-processor program when $P = 4$ and when $P = 16$. Should the answer be the same in all these cases?
6. Compare the memory requirement per processor for the single-processor program and the multi-processor program when $n \gg 1$.
7. How many floating point operations are needed to generate the vector $\underline{v}$?
   How many floating point operations are needed to compute S_n in (2)?
   Is the multi-processor program load-balanced?
8. Do you think parallel processing is attractive to use for solving this problem?