TMA4280: Introduction to Supercomputing

Problem set 4

Spring 2014

NOTE THAT:

- this problem set is mandatory;
- you can work on this problem in groups with up to 3 members;
- a report describing the solution should be written (as continous text);
- the source code should be handed in with the report (do not attach it to the report);
- please make sure that you have answered all the questions;
- the due date is Monday, February 17, 2014;
- the report will count 10% towards the final grade.

Department of Mathematical Sciences
NTNU, N-7491 Trondheim, Norway

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 $k = 3, \ldots, 14$;
   - print out the difference $S - S_n$ in double precision.
2. Do the necessary changes to utilize shared memory parallelization through OpenMP.
3. Write a program to compute the sum S_n using P processors where P is a power of 2, and a distributed memory model (MPI). The program should work as follows: Only processor 0 should be responsible for generating the vector elements. Processor 0 should partition and distribute the vector elements evenly among all the processors. Each processor should be responsible for summing up its own part. At the end, all the partial sums should be added together and made available on processor 0 for printout. Report the difference $S - S_n$ in double precision for different values of n.
4. Confirm that your program also works when you are using OpenMP/MPI in combination.
5. Which MPI calls are convenient/necessary to use?
6. Compare the difference $S - S_n$ from the single-processor program and the multi-processor program when $P = 2$ and when $P = 8$. Should the answer be the same in all these cases?
7. Compare the memory requirement per processor for the single-processor program and the multi-processor program when $n \gg 1$.

8. 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?

9. Do you think parallel processing is attractive to use for solving this problem?