

CSCE 689-701 Summer 2020

HW 1: Parallel Programming with MPI

Due: 11:59pm Monday, July 6, 2020

Compile and execute the program in the file `compute_pi_mpi.c`, which computes an estimate of π using the parallel algorithm discussed in class. The program is available on the shared Google Drive for this class. It should be compiled and executed on either `ada.tamu.edu` or `terra.tamu.edu`.

Load the Intel software stack prior to compiling and executing the code.

```
module load intel/2017A
```

To compile, use the command:

```
mpicc -o compute_pi_mpi.exe compute_pi_mpi.c
```

To execute the program, use

```
mpirun -np <p> ./compute_pi_mpi.exe <n>
```

where `<n>` represents the number of intervals and `<p>` represents the number of processes.

The output of a sample run is shown below.

```
mpirun -np 4 compute_pi_mpi.exe 100000000
```

```
n = 100000000, p = 4, pi = 3.1415926535897749, relative error =  
5.80e-15, time (sec) = 0.0608
```

The run time of the code should be measured when it is executed in dedicated mode. Use the batch file `compute_pi_mpi.job`, to execute the code in dedicated mode using the following command on ADA:

```
bsub < compute_pi_mpi.job
```

On Terra, you will need to use `compute_pi.terra_job`, and the corresponding command is:

```
sbatch compute_pi.terra_job
```

Execute the code for $n=10^8$ with p chosen to be 2^k , for $k = 0, 1, \dots, 6$. Specify `ptile=4` in the job file. Using the experimental data obtained from these experiments, answer the following questions.

1. (20 points) Plot execution time versus p to demonstrate how time varies with the number of processes. Use a logarithmic scale for the x-axis.
2. (20 points) Plot speedup versus p to demonstrate the change in speedup with p .
3. (10 points) Using the definition: $\text{efficiency} = \text{speedup}/p$, plot efficiency versus p to demonstrate how efficiency changes as the number of processes is increased.
4. (10 points) What value of p minimizes the parallel runtime?
5. (20 points) With $n=10^9$ and $p=64$, determine the value of `ptile` that minimizes the `total_time`. Plot time versus `ptile` to illustrate your experimental results for this question.

6. (20 points) Repeat the experiments with $p=64$ for $n=10^2, 10^4, 10^6$ and 10^8 .
- Plot the speedup observed w.r.t. $p=1$ versus n .
 - Plot the relative error versus n to illustrate the accuracy of the algorithm as a function of n .

Submission: Upload a single PDF or MSWord document with your answers to Canvas.