

CS 566 Parallel Processing, Spring 2006

Programming Assignment 1

Assigned: February 8

Due: March 6, 11:59pm

1 Goals

In this assignment, you will study various aspects of parallel programming using MPI. The well-known matrix multiplication (MM) problem which is fundamental to many high-performance applications will be considered. You will be using the ARGO-NEW parallel cluster provided by ACCC as the experimental testbed. Your goal is to solve $A(X^k)A^T$, where

- A is a $1 \times n$ matrix
- X is a $n \times n$ matrix
- k is an input parameter

2 Experiment

1. Identify a set of different mappings (parallel formulations) of the MM problem, such as those that vary how to decompose the input data, how to decompose the computation workload, the choice of MPI calls used, etc.. Let \mathcal{S} denote the set containing each of these parallel formulations.
2. You need to design the code for each of the parallel formulations you choose.
3. Experiment the *timing*, *speedup*, and *efficiency* variations by varying the different parameters along the lines suggested here. Note that these are only suggestions: the actual parameter ranges you vary will depend on various factors such as the workload on and the availability of the ARGO-NEW nodes, OS limits etc..

n , **size of matrix:** $n = 2^6, 2^8, 2^{10}, 2^{12}, 2^{14}, \dots$

t , **number of tasks:** Choose an appropriate range for t

p , **number of processors:** $n = 1, 2, 4, 8, 16, 32, 64$

k , **problem input parameter:** $k = 1, 2, 3, 4, \dots$

3 Input and Testing

- While you debug your code, please work with small data sets, few processes, and few processors i.e., small n , small k , small t , small p .
- The matrix A and X have numbers taken from the range $\{-1, 0, 1\}$.
- **Input Method 1:** The input is 2 numbers: the probability of -1, and the probability of +1. Create the matrices at the root process using the given probabilities. *This matrix is created at the root process only. You cannot use this formula to create the matrix at the worker processors.*
- **Input Method 2:** The input is taken at the keyboard as follows. Take in a sequence of 4 numbers from the range $\{-1, 0, 1\}$. For example, the sequence $[-1, 0, +1, 0]$.

- A is this sequence repeated $n/4$ times.
- Row i of matrix X is A after a cyclic-right- f -shift of A , where $f = \sum_{k=1}^i k$.

This matrix is created at the root process only. You cannot use this formula to create the matrix at the worker processors.

4 Output

1. Put all your files including Makefile in directory `assignment1`. Turnin all the code, including the makefile, by executing:

```
turnin -c cs566 -p assignment1 assignment1
```
2. Submit a hard copy of all the files including the code file.
3. Submit a hard-copy report describing your experiment. The report must have the following sections/information.
 - (a) **Formulations:** Describe each of the formulations with in the set \mathcal{S} in about 1-2 pages each. Justify your selection of this formulation. Include the list of MPI calls you used in the implementation of each formulation.
 - (b) **Parameter ranges:** For each of the formulations describe the range of parameters you experimented with.
 - (c) **Results:** Give the tables and plot the graphs showing the *timing*, *speedup*, and *efficiency* variations for each of the formulations, for the range of parameters you used. Remember to use appropriate scales for the graphs.
 - (d) **Analysis:** Analyse the results. Give your interpretation and reading of the results for each formulation in \mathcal{S} , and across formulations in \mathcal{S} , i.e., also compare the results of the different formulations. You should address questions such as the following sample list:
 - Which formulation is better? under what (or all) circumstances? Why?
 - How and why does efficiency vary the way it does?
 - How and why does speedup vary the way it does?
 - Can you measure the computation time versus communication time? How do the formulations compare with each other with respect to this breakup of the time overhead?
 - Can the communication overhead be overlapped with your computation overhead? Do your formulations/algorithms do so?

Explain why you observe what you observe. In particular, any anomalous observations should be explained.

- (e) **Lessons:** What insights you learned from this assignment.

5 Grading

The problem is reasonably well-formulated but the experimental approach is open-ended. Your goal is to get the most parallelism in solving the MM problem. The depth of your choice of parallel formulations you choose to experiment with will be judged. Your assignment will also be judged on how *comprehensively* and *methodically* you have designed and run the experiment, and reported on the results. Your insights into the analysis of the results will also be considered in judging the assignment.

6 Reference Material/Chapters

Chapters 3, 6, and 8 from the text. The sample programs of the ACCC web site.