Speeding Up via MPI and MapReduce

Date assigned: Week 14 (April 22, 2024)
Date due: Sunday, **May 12**, 2024, at 11:59 midnight. (No late submission allowed.)
Submission at: Your section's GCR
Max points: 5000 points
Collaboration: This is a team project with 5 members. Collaboration is only allowed among the team members.

Note: Please carefully read the full assignment first before starting.

This assignment aims to allow you to speed up a serial code using MPI and MapReduce.

What to submit:

(1) You will need to provide us with a report that will journal your journey on how you solved the problems posed in this assignment, along with the specific answers to the questions.
(2) Your code with a README that will tell how to execute your code

We will continue with our matrix multiplication example from the first and second assignments. You need to follow the input and output rules as we laid them in the first assignment.

This is a group assignment. You need to make a group of 5 students in total.

Part 1: [1000 points] Parallelize your c-based matrix multiplication code (from assignment 1) using MPI. You can keep your SIMD instructions enabled. You will progressively increase your machines by utilizing the laptops/computers of your teammates and get a table as follows:

Number of compute nodes used	Execution Time in seconds	Speed-up
1		—
2		
3		
4		
5		

The common crawl (https://data.commoncrawl.org/crawl-data/CC-MAIN-2024-10/index.html) regularly crawls the WWW. We will utilize the 2024 crawl (https://data.commoncrawl.org/crawl-data/CC-MAIN-2024-10/index.html). Data is in three different formats (see https://commoncrawl.org/blog/web-archiving-file-formats-explained for details of these formats.) We will use the WET format to get website text data only.

Each member of your group will pick 5 randomly chosen WET files from the above-mentioned crawled data (so there will be 25 files in total for a group).

Install open-source Hadoop such that your five computers become part of a Hadoop cluster. Put the 25 WET files in your cluster's HDFS file system. Now you need to process them on your local 5-node cluster using MapReduce.

Part 2: [2000 points] Your task for MapReduce is word counting of English words (the data might have text from other languages such as German, Spanish, etc. You should ignore all such data in your files). Report the word-counting result for your group on your 25 WET files. Also, tell us how much time it took for the MapReduce job to finish.

Part 3: [2000 points] For this part, you will use the same 25 files used in the previous part. Now, your MapReduce task is to generate a reverse index where you find out which word is in which webpage's URL. This task will have an offline part where you use your local MapReduce cluster to generate a reverse index. An online part where you use one of your 5 machines to load the MapReduce results into a hash table and provide a front end (or command line interface) for us to query it.

Once processed using MapReduce, you will use a hash map to answer the queries. In a query, we will ask you to search a word in your index, and if found, you will tell on which web pages it is present. If it is not in your data, you will say not found. You can ignore words like "to", "the" etc.

You should have a substantial amount of your GCP free credits remaining after this assignment. We encourage you to use them well to learn what interests you on GCP (probably after the semester ends and you have tons of time). You might like to try GCP data proc to use hosted MapReduce. You might find out something interesting to work on for your FYP during this exercise.