IMPLEMENT COLLECTIVE COMMUNICATION OPERATION

Homework #5

By

John Robertson

CS 432 Parallel Computing  
April 23, 2021

Problem Specification:

**The allgather communication algorithm is implemented by implementing and pairing gather and broadcast.**

Program Design:

**The allgather function is a mashup of my understanding of the gather algorithm as given in the MPI docs' high-level description (not the source code) of how MPI\_Gather is implemented and how Bangalore described how broadcast is implemented. Dr. Bangalore mentioned that a blocking operation like MPI\_Recv can be used when it's at a point where it would otherwise be waiting.**

Testing Plan:

**For testing, the driver used for the provided myscatter.c was used in a Python script so 240 runnings of the program could be generated. The Python script for testing is as follows:**

***#!/bin/bash***

***import subprocess***

***for k in range(5):***

***for j in range(5, 21):***

***acc = 0.0***

***for i in range(3):***

***acc += float(subprocess.check\_output(***

***f"mpiexec -np {2\*\*k} ./a.out {2\*\*j}",***

***shell=True***

***))***

***acc /= 3***

***print(f"Average time taken for {2\*\*k} processes, {2\*\*j} bytes: {acc}")***

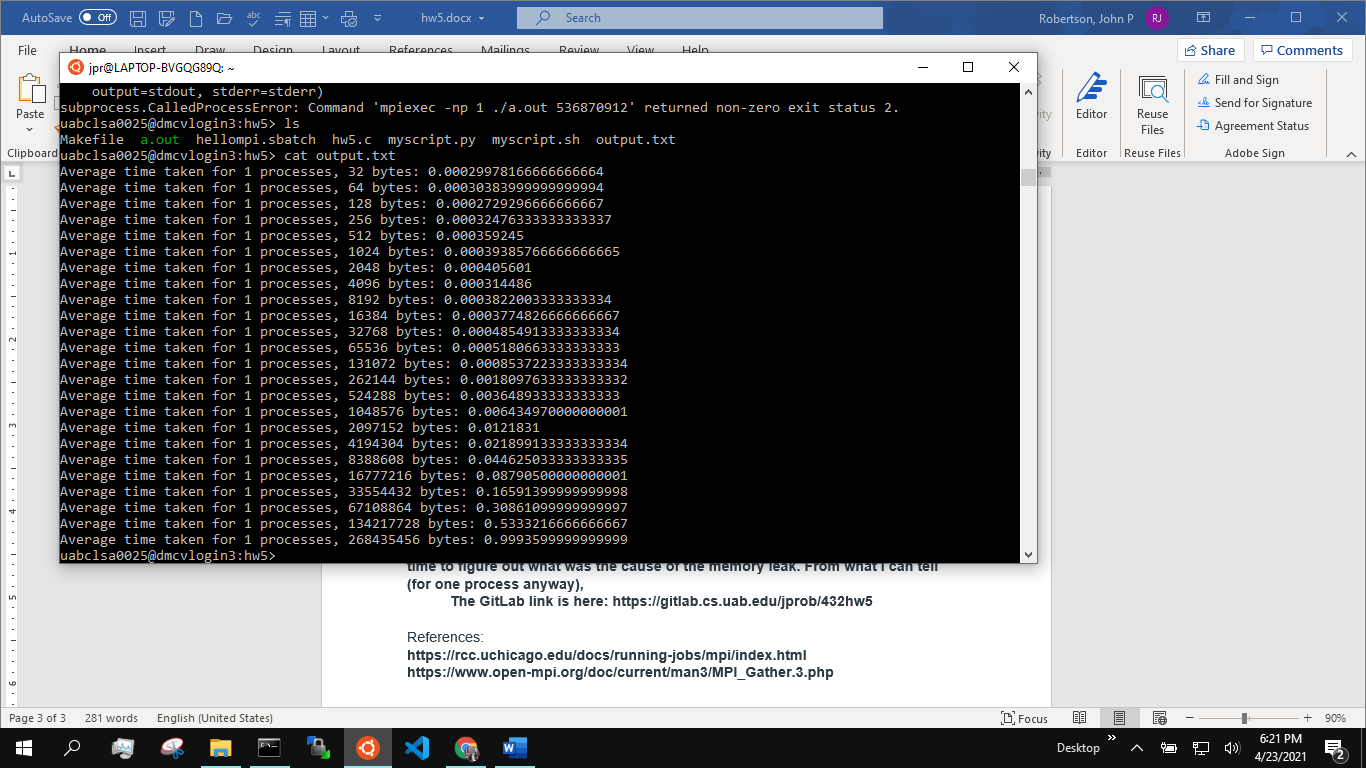
***print("")***

**Python was used for the script rather than bash because bash does not natively support floating point arithmetic.**

Test Cases:

**The grid of what test cases were used, 2^5 through 2^20 bytes computed over 1 process, is shown below. An error who cause was never discovered led to it never working for more than one process, so their results are not available.**

Analysis and Conclusions:

**The error message that I kept getting and was unable to fix indicated that if there was a second process, some memory leak would happen. I did not have the time to figure out what was the cause of the memory leak. From what I can tell (for one process anyway), the increase in time is exponential; running it with bytes up to 2^30 seem to indicate this as well:**

**The GitLab link is here: https://gitlab.cs.uab.edu/jprob/432hw5**

References:

**https://rcc.uchicago.edu/docs/running-jobs/mpi/index.html**

**https://www.open-mpi.org/doc/current/man3/MPI\_Gather.3.php**