CS 633A Assignment 3

Gautam Chauhan, Sreejit Bose (20111020, 20111065)

gautamc20, sreejit20@iitk.ac.in

April 18, 2021

Code takes data file name as argument.
File format:

- src.c - Source code file
- run.py - Script to run the code
- makefile - To compile the source code
- script.py - Script to generate hostfile
- nodefile.txt - List of nodes used by script.txt
- output.txt - Output is stored here
- plot_1.png - Plot using 1 node
- plot_2.png - Plot using 2 nodes

Data distribution strategy:

1. After reading the csv file using process 0 we put each column in a array using column major order.
2. Numbers of columns to needed to be distributed to each node. First we evenly divide columns and left over columns are given to a node in increasing order of rank until all columns are distributed.
   eg. if we have 41 columns and 4 nodes then columns distributed are 11,10,10,10 and if columns are 43 then 11,11,11,10
3. These are columns are distributed using MPI Scatterv as data is not evenly distributed in every case.
4. When each process have finished calculating minimun for each row then rank 0 collect data using MPI Gatherv.
5. Process 0 then calculates the overeall minimum across all stations and all years.

Code description:

- Initially, before reading the file, number of rows and columns are calculated.
- Data is stored as a single array in column major order, so every row of a column is together.
- Numbers of columns to needed to be distributed to each node. First we evenly divide columns and left over columns are given to a node in increasing order of rank until all columns are distributed.
- Above step is only done by rank 0, then information a total rows, total column and number of column a node will receive is distributed using scatter.

- Using Scatterv all columns are distributed to nodes.
- Nodes then calculate minimum temperature for each columns and save it an array.
- These arrays are bought back to rank 0 using gatherv in order to save them in a output.txt and calculate overall minimum across all stations and all years.

Observations:

From the plots in figure 1 and figure 2, it can be seen that if we increase the number of processes the code takes more time for number of nodes 1 and number of nodes 2 as well. Moreover, upon increasing the number of nodes from 1 to 2 the time increases.
This indicates communication time between processes and nodes taking more time than the calculation itself.

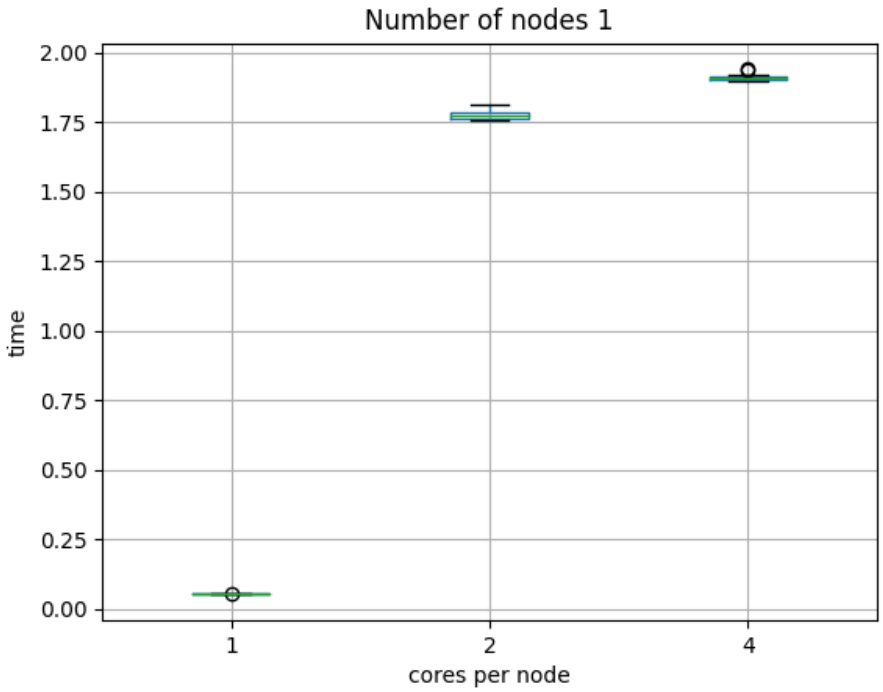

Figure 1: Time Taken for number of nodes = 1

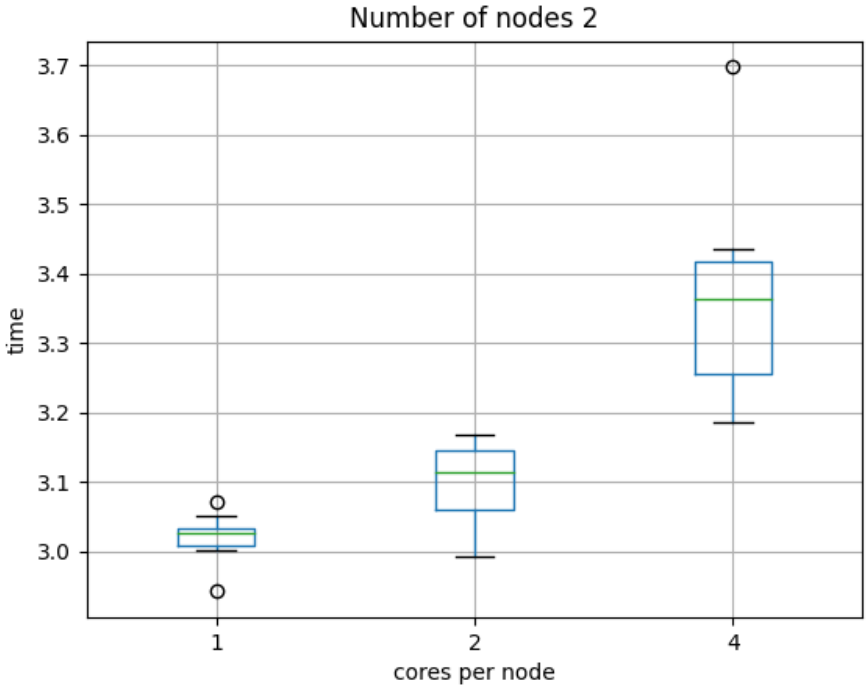

Figure 2: Time Taken for number of nodes = 2