Parallel String Match

ROHIT BAL

CSE 633 FALL 2010

INSTRUCTOR: DR. RUSS MILLER

A Quick Recap…..

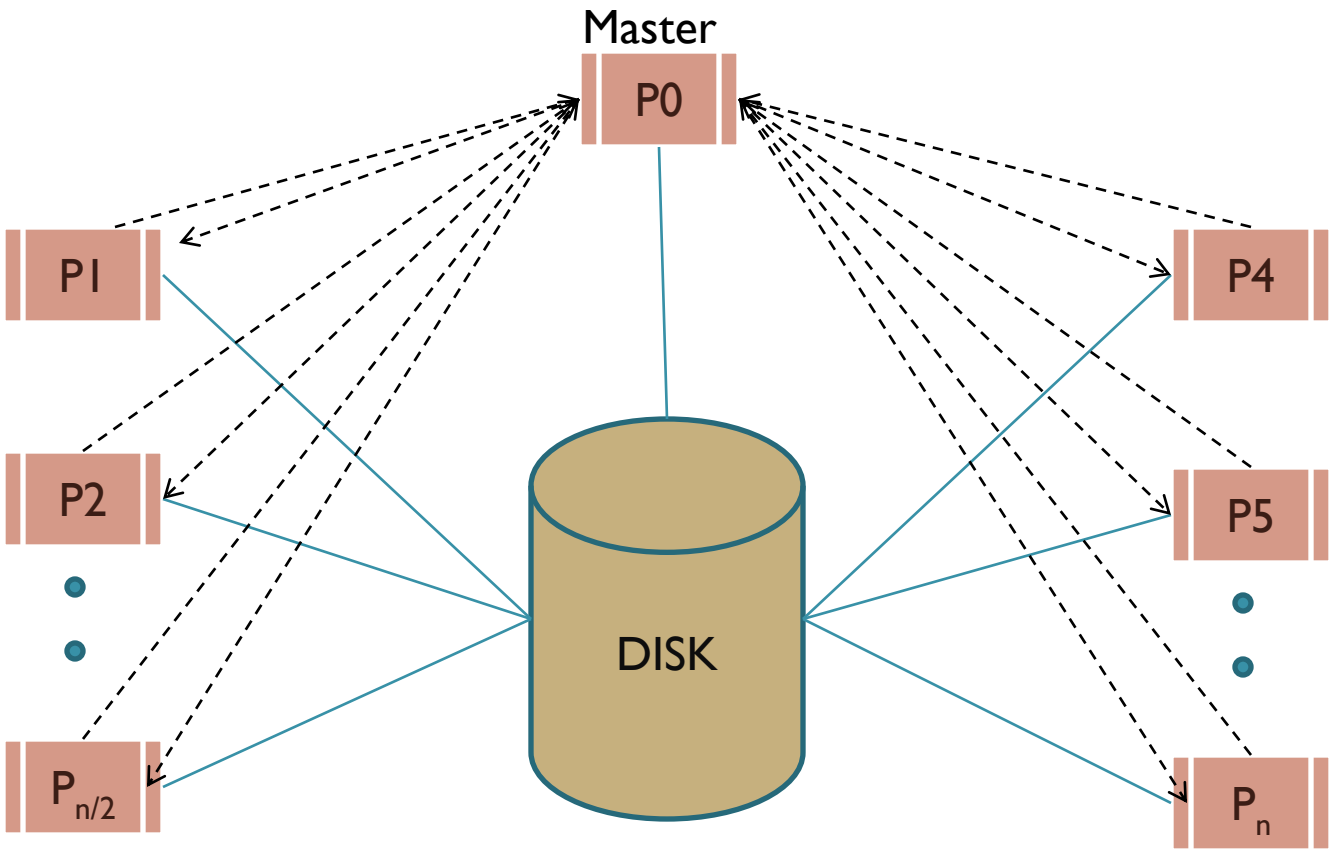
- Given a pattern string *P* , find all lines in all files matching *P* from a given dataset of *M* files using *N* processors.

- All files are text files

System Overview

- Master – Worker model
- The Master node is just the coordinator
- The Workers do the actual search
- Data set is located on a shared disk

System Architecture



Sequential Search

- Rudimentary approach
- Loop through all files in the directory
- Fetch a line from each file and do a search
- Print/Store results

Parallel Search

- Complex Code
- The directory is opened on each worker PE
- Each worker PE searches on a fraction of the total number of files
- The results are printed to STDOUT

Implementation

- Accept the pattern to be found as command line argument
- PE with rank 0 is the Master
- The Master sends the pattern to all workers using MPI_Send()
- All other PEs are workers
- They receive the pattern using MPI_Receive()

Implementation (continued)

- Each worker opens the directory using the DIR * pointer
- *The worker then 'rewinds' the DIR and jumps to the first file in its loop*
- The 'TYPE' member of DIR is used to identify files from directories
- A *for* loop searches on a fixed number of files
- Results are printed to STDOUT

Load Balancing

- Larger files need load balancing
- Synchronous vs. Asynchronous calls
- Point of entry problems with asynchronous calls
- Each chunk has to wait in queue until the current set of files has been processed

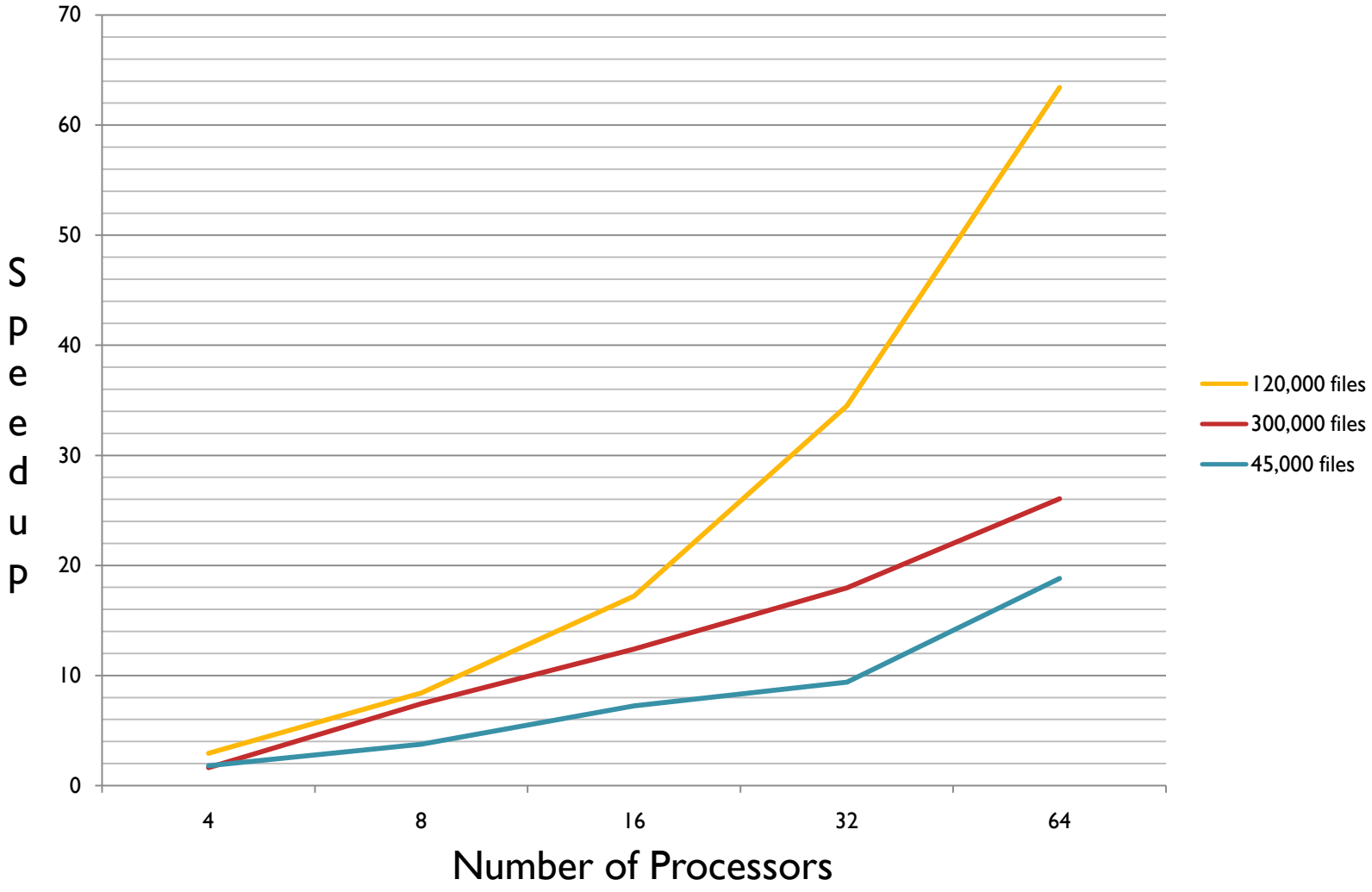
Possible solutions

- Allow the Master to take over
- Some interrupt to running process
- Multithreading : seems like the best approach

Results for Sequential Search

Number of Files	Search Time (seconds)
45,000 (~3GB)	276
120,000 (~6GB)	380
300,000 (~11GB)	521

Parallel Search Results



Observations

- Linear Speedup seems irrelevant of Data size
- Network delay did not seem to be a factor (possibly due to Myrinet, Infiniband)

Future Work

- Load Balancing using Multithreading
- Test for much larger data
- Test for a different architecture

Thank You!

Any Questions?