Project Design: Distributed Computing using pi

Harsh Patil
Ganesh Rajasekharan
Shikha Soni

Overview

Deliverables

- Master work distributer
- Slave
- Parts merge program
- Compilation and execution script

Main Goal

To perform a distributed task of sorting and adding alphanumeric values while introducing some fault tolerant mechanism in the design

Design- 1

Proposal: A parallel computational design

- Sends chunk of data to the salves and gets the sorted chunks back
- At master combines the slave input into more chunks for a second pass to the slaves
- Run until only two parts remain, merge them on master

Pros or Cons

- Utilizes the free memory at slave to perform some part of merge
- Increases network I/O for higher passes (each chunk increases in size)

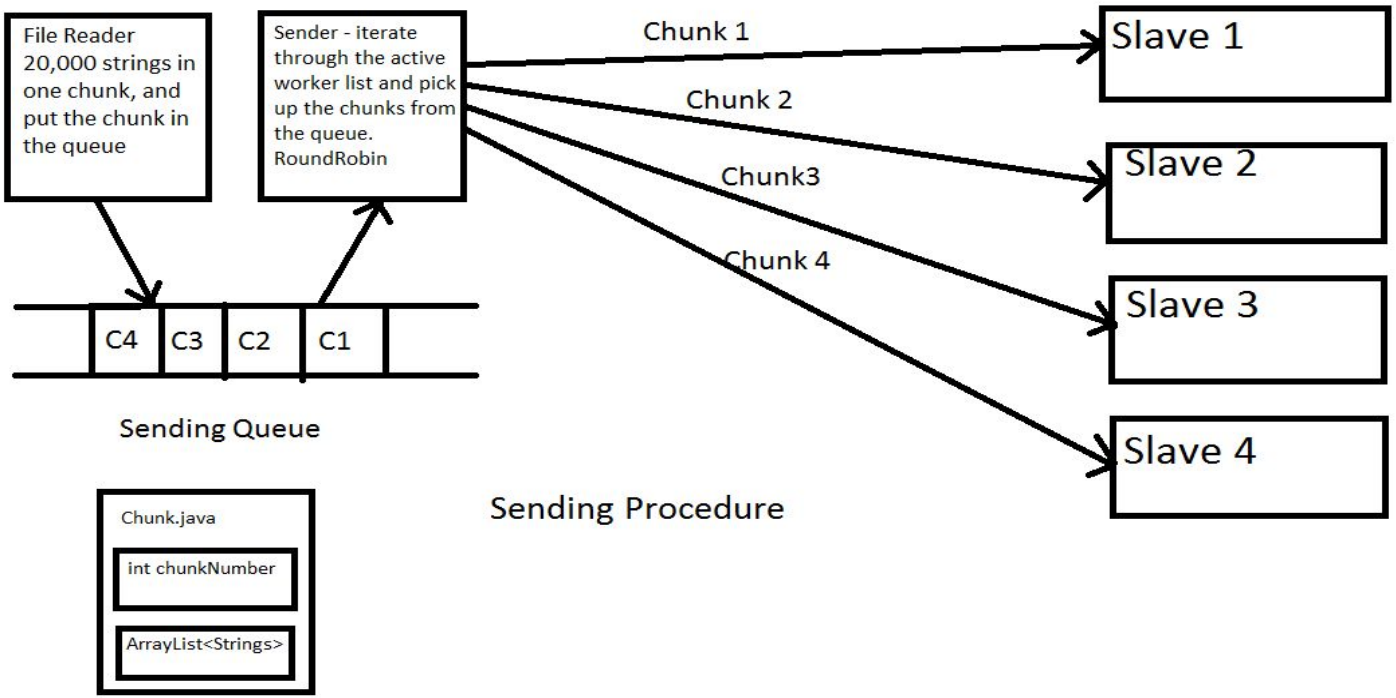
Design- 2

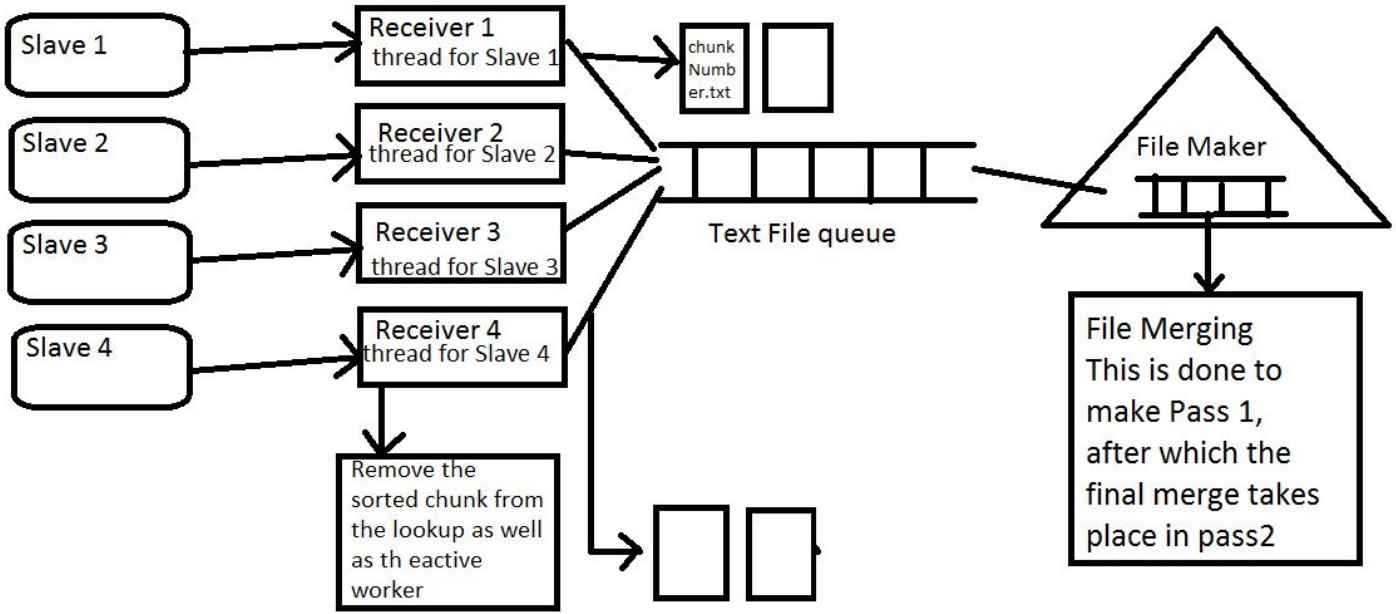
Proposal: Compute once design

- Sends chunk of data to the slaves and gets the sorted chunks back
- At master combines the slave input into larger sorted chunks and saves them on disk
- Run the merging step of the files on the master

Pros or Cons

- Reduces network I/O, gets the sorted files back in quick time
- Increases file r/w
- Need to synchronize the combining and of slave input and merging of files
- Heap size increases

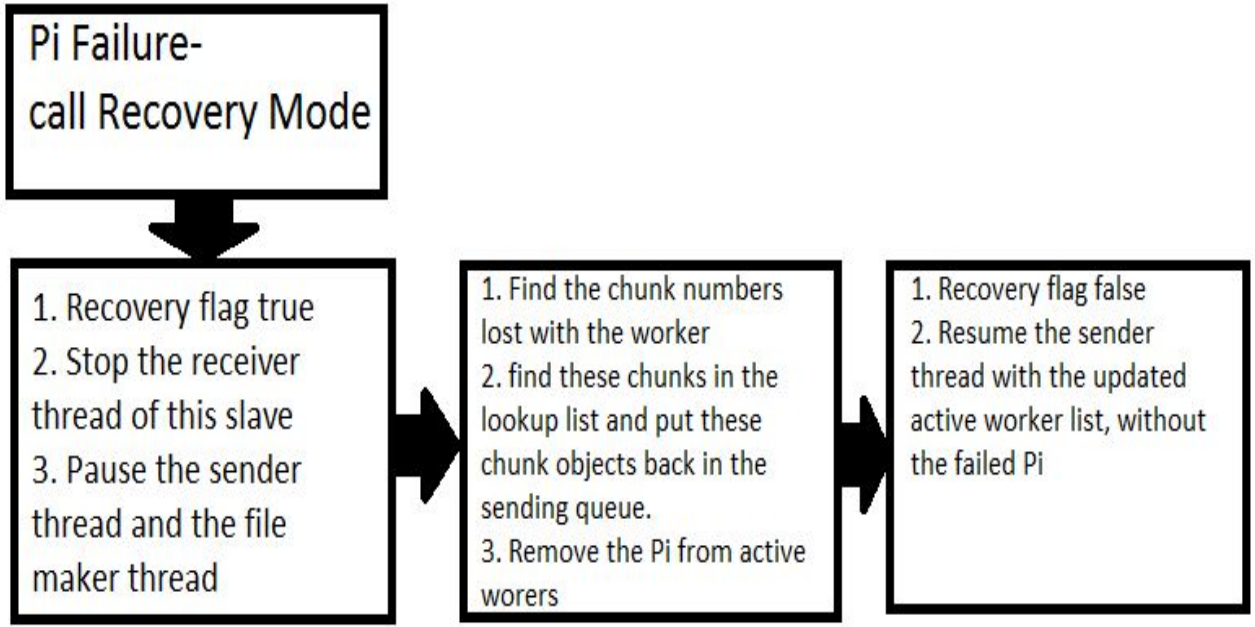

File Reader 20,000 strings in one chunk, and put the chunk in the queue
Sender - iterate through the active worker list and pick up the chunks from the queue. RoundRobin
Chunk 1
Chunk 2
Chunk3
Chunk 4
Slave 1
Slave 2
Slave 3
Slave 4
C4
C3
C2
C1
Sending Queue
Chunk.java
int chunkNumber
ArrayList<Strings>
Sending Procedure



Receiving Procedure

Fault Tolerance

- Each chunk has a number and content
- Remember chunks distributed amongst the slaves.
- Look up list and Active worker table
- Pi failure causes the system’s recovery mode on
- Redistribute back the chunks
- Remove pi from the active worker list


Pi Failure-
call Recovery Mode
1. Recovery flag true
2. Stop the receiver thread of this slave
3. Pause the sender thread and the file maker thread
1. Find the chunk numbers lost with the worker
2. find these chunks in the lookup list and put these chunk objects back in the sending queue.
3. Remove the Pi from active worers
1. Recovery flag false
2. Resume the sender thread with the updated active worker list, without the failed Pi

Implementation

Final Design:

- Compute Once Design
- Merge sorted merged files into the final file.
- Merge done by buffering and reading from all the files to be merged.

Why is it better?

- If one pi fails the design 1 will cost additional data loss and data transfer.

The changes observed in the time taken with changes in the file size

Design 1

File Size (MB)	Number of chunks	Time to complete (se
5	1000	700
5	5000	510
5	10000	505
5	15000	498
5	20000	515
5	30000	522
10	1000	1244
10	5000	1212
10	10000	1186
10	15000	1156
10	20000	1104
10	30000	1023
19	20000	46521